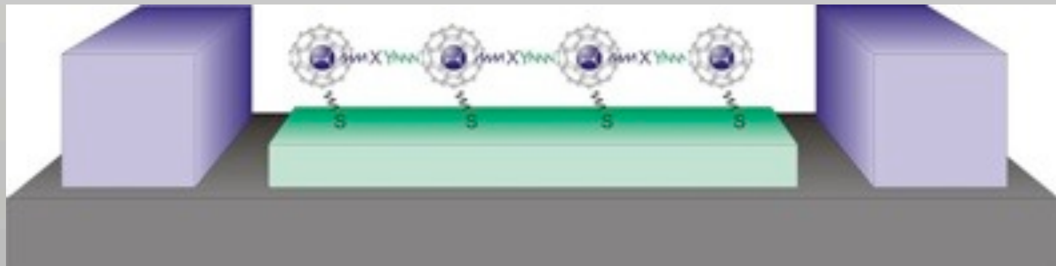
lecture WS ’15

Quantum Computers
- how to make them work -
Implementation of the Deutsch-Jozsa and of the Grover algorithm

C. Meyer, C.M. Schneider



quantum parallelism

input

$$\left\{ \begin{array}{c} a_1 \, |00> \\ + \\ a_2 \, |01> \\ + \\ a_3 \, |10> \\ + \\ a_4 \, |11> \end{array} \right\}$$

- Superposition for input created by Hadamard gates

quantum parallelism

input

$$\left\{ \begin{array}{c} a_1 |00\rangle \\ + \\ a_2 |01\rangle \\ + \\ a_3 |10\rangle \\ + \\ a_4 |11\rangle \end{array} \right\} \xrightarrow{F} \left\{ \begin{array}{c} a_1 F |00\rangle \\ + \\ a_2 F |01\rangle \\ + \\ a_3 F |10\rangle \\ + \\ a_4 F |11\rangle \end{array} \right\}$$

- Superposition for input created by Hadamard gates
- Functions are represented by unitary operators

quantum parallelism

input | output

$$\left\{ \begin{array}{c} a_1\,|00\rangle \\ + \\ a_2\,|01\rangle \\ + \\ a_3\,|10\rangle \\ + \\ a_4\,|11\rangle \end{array} \right\} \xrightarrow{F} \left\{ \begin{array}{c} a_1\,F\,|00\rangle \\ + \\ a_2\,F\,|01\rangle \\ + \\ a_3\,F\,|10\rangle \\ + \\ a_4\,F\,|11\rangle \end{array} \right\} = \left\{ \begin{array}{c} b_1\,|00\rangle \\ + \\ b_2\,|01\rangle \\ + \\ b_3\,|10\rangle \\ + \\ b_4\,|11\rangle \end{array} \right\}$$

- Superposition for input created by Hadamard gates
- Functions are represented by unitary operators
- Quantum state tomography – but how to get a "real" result?

Deutsch algorithm: table of truth

D. Deutsch, *Proceedings of the Royal Society of London* **A 400** (1985), 97

simplest example: 1-bit-to-1-bit function f : x $\rightarrow$ {0,1}

Deutsch algorithm: table of truth

D. Deutsch, *Proceedings of the Royal Society of London* **A 400** (1985), 97

simplest example: 1-bit-to-1-bit function $f : x \rightarrow \{0,1\}$

$f(0)$
$f(1)$

Deutsch algorithm: table of truth

D. Deutsch, *Proceedings of the Royal Society of London* **A 400** (1985), 97

simplest example: 1-bit-to-1-bit function $f : x \rightarrow \{0,1\}$

	case 1
$f(0)$	0
$f(1)$	0

Deutsch algorithm: table of truth

D. Deutsch, *Proceedings of the Royal Society of London* **A 400** (1985), 97

simplest example: 1-bit-to-1-bit function $f : x \rightarrow \{0,1\}$

	case 1	case 2
$f(0)$	0	1
$f(1)$	0	1

Deutsch algorithm: table of truth

D. Deutsch, *Proceedings of the Royal Society of London* **A 400** (1985), 97

simplest example: 1-bit-to-1-bit function f : x → {0,1}

	case 1	case 2	case 3
$f(0)$	0	1	0
$f(1)$	0	1	1

Deutsch algorithm: table of truth

D. Deutsch, *Proceedings of the Royal Society of London* **A 400** (1985), 97

simplest example: 1-bit-to-1-bit function $f : x \rightarrow \{0,1\}$

	case 1	case 2	case 3	case 4
$f(0)$	0	1	0	1
$f(1)$	0	1	1	0

Deutsch algorithm: table of truth

D. Deutsch, *Proceedings of the Royal Society of London* **A 400** (1985), 97

simplest example: 1-bit-to-1-bit function $f : x \rightarrow \{0,1\}$

	constant			
	case 1	case 2	case 3	case 4
$f(0)$	0	1	0	1
$f(1)$	0	1	1	0

Deutsch algorithm: table of truth

D. Deutsch, *Proceedings of the Royal Society of London* **A 400** (1985), 97

simplest example: 1-bit-to-1-bit function $f : x \rightarrow \{0,1\}$

	constant		balanced	
	case 1	case 2	case 3	case 4
$f(0)$	0	1	0	1
$f(1)$	0	1	1	0

Deutsch algorithm: table of truth

D. Deutsch, *Proceedings of the Royal Society of London* **A 400** (1985), 97

simplest example: 1-bit-to-1-bit function $f : x \rightarrow \{0,1\}$

	constant		balanced	
	case 1	case 2	case 3	case 4
$f(0)$	0	1	0	1
$f(1)$	0	1	1	0

Deutsch algorithm: table of truth

D. Deutsch, *Proceedings of the Royal Society of London* **A 400** (1985), 97

simplest example: 1-bit-to-1-bit function $f : x \rightarrow \{0,1\}$

	constant		balanced	
	case 1	case 2	case 3	case 4
$f(0)$	0	1	0	1
$f(1)$	0	1	1	0

real or false?

Deutsch algorithm: table of truth

D. Deutsch, *Proceedings of the Royal Society of London* **A 400** (1985), 97

simplest example: 1-bit-to-1-bit function $f : x \rightarrow \{0,1\}$

	constant		balanced	
	case 1	case 2	case 3	case 4
$f(0)$	0	1	0	1
$f(1)$	0	1	1	0

real or false?

Deutsch algorithm: table of truth

D. Deutsch, *Proceedings of the Royal Society of London* **A 400** (1985), 97

simplest example: 1-bit-to-1-bit function $f : x \rightarrow \{0,1\}$

	constant		true	
	case 1	case 2	case 3	case 4
$f(0)$	0	1	0	1
$f(1)$	0	1	1	0

real or false?

Deutsch algorithm: table of truth

D. Deutsch, *Proceedings of the Royal Society of London* **A 400** (1985), 97

simplest example: 1-bit-to-1-bit function $f: x \rightarrow \{0,1\}$

	constant		true	
	case 1	case 2	case 3	case 4
$f(0)$	0	1	0	1
$f(1)$	0	1	1	0




real or false?

Deutsch algorithm: table of truth

D. Deutsch, *Proceedings of the Royal Society of London* **A 400** (1985), 97

simplest example: 1-bit-to-1-bit function f : x → {0,1}

	false		true	
	case 1	case 2	case 3	case 4
$f(0)$	0	1	0	1
$f(1)$	0	1	1	0




real or false?

Deutsch algorithm: table of truth

D. Deutsch, *Proceedings of the Royal Society of London* **A 400** (1985), 97

simplest example: 1-bit-to-1-bit function $f: x \rightarrow \{0,1\}$

	false		true	
	case 1	case 2	case 3	case 4
$f(0)$	0	1	0	1
$f(1)$	0	1	1	0
				
$f(0) \oplus f(1)$	0	0	1	1

real or false?

strength of quantum parallelism

- calculate a function f(x) with $x \in \{0,\ldots,K\}$ for all x at once

strength of quantum parallelism

- calculate a function f(x) with $x \in \{0,\ldots,K\}$ for all x at once

classical

```
function balanced(f:function)
a=f(0)
b=f(1)
return a ⊕ b
```

strength of quantum parallelism

- calculate a function f(x) with $x \in \{0,\ldots,K\}$ for all x at once

<u>classical</u>

function balanced(**f**:function)
a=**f**(0)
b=**f**(1)
return $a \oplus b$

<u>quantum</u>

function balanced(U_f:function)
$|\Psi\rangle_1 = H\,|\Psi\rangle_0 = H\,|0\rangle|1\rangle$
$|\Psi\rangle_2 = U_f\,|\Psi\rangle_1$
$|\Psi\rangle_3 = H_x\,|\Psi\rangle_2 = |f(0)\oplus f(1)\rangle|R\rangle$
return $|f(0)\oplus f(1)\rangle$

strength of quantum parallelism

- calculate a function f(x) with $x \in \{0,\dots,K\}$ for all x at once

<u>classical</u>

function balanced(**f**:function)
a=**f**(0)
b=**f**(1)
return $a \oplus b$

<u>quantum</u>

function balanced($\mathbf{U_f}$:function)
$|\Psi\rangle_1 = H\,|\Psi\rangle_0 = H\,|0\rangle|1\rangle$
$|\Psi\rangle_2 = \mathbf{U_f}\,|\Psi\rangle_1$
$|\Psi\rangle_3 = H_x\,|\Psi\rangle_2 = |f(0)\oplus f(1)\rangle|R\rangle$
return $|f(0)\oplus f(1)\rangle$

- use quantum computing only if f(x) is "expensive" to calculate (time, memory), e. g., spin dynamics for huge structures

strength of quantum parallelism

- calculate a function f(x) with $x \in \{0,\ldots,K\}$ for all x at once

<u>classical</u>

function balanced(**f**:function)
a=**f**(0)
b=**f**(1)
return $a \oplus b$

<u>quantum</u>

function balanced($\mathbf{U_f}$:function)
$|\Psi\rangle_1 = H\,|\Psi\rangle_0 = H\,|0\rangle|1\rangle$
$|\Psi\rangle_2 = \mathbf{U_f}\,|\Psi\rangle_1$
$|\Psi\rangle_3 = H_x\,|\Psi\rangle_2 = |f(0)\oplus f(1)\rangle|R\rangle$
return $|f(0)\oplus f(1)\rangle$

- use quantum computing only if f(x) is "expensive" to calculate (time, memory), e. g., spin dynamics for huge structures

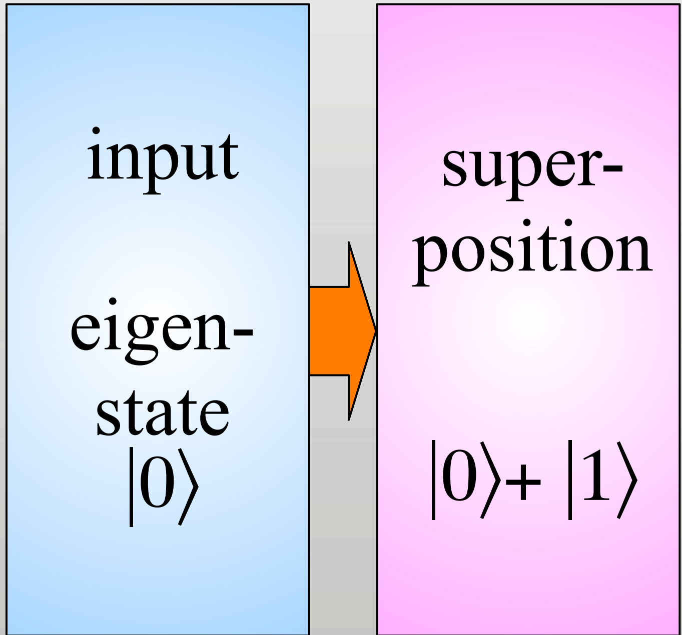
- problem: superposition cannot be read-out: it will always collapse to an eigenstate

Deutsch algorithm: principle

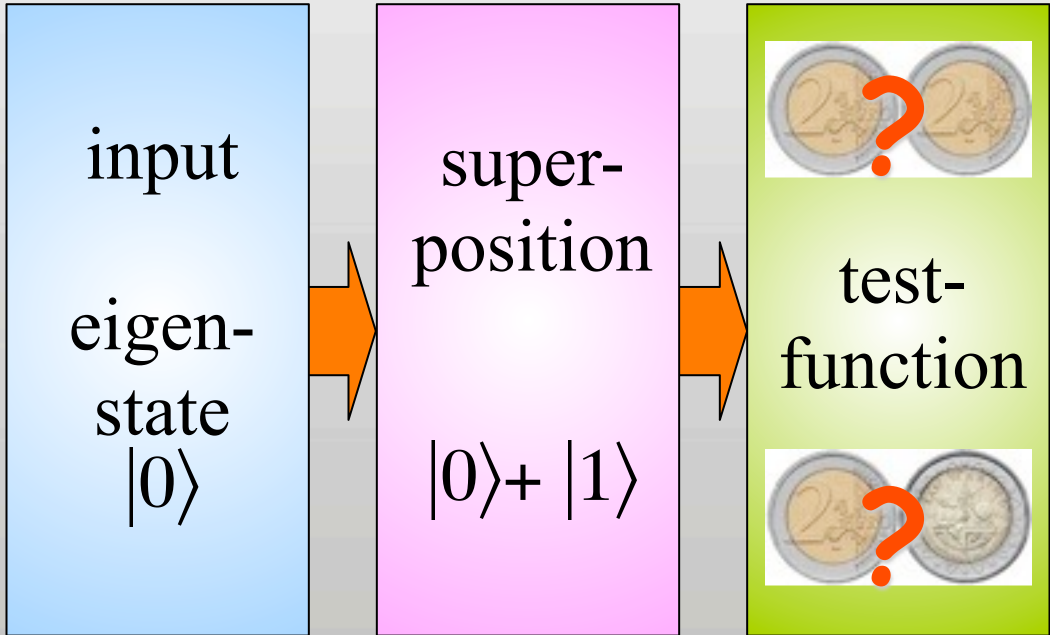
input

eigen-
state
$|0\rangle$

Deutsch algorithm: principle



Deutsch algorithm: principle



Deutsch algorithm: principle

input

eigen-state

$|0\rangle$

→

super-position

$|0\rangle + |1\rangle$

→

test-function

→

transfer to eigen-state

$|0\rangle$ **or** $|1\rangle$

Deutsch algorithm: principle

input

eigen-state $|0\rangle$

→

super-position

$|0\rangle + |1\rangle$

→

test-function

→

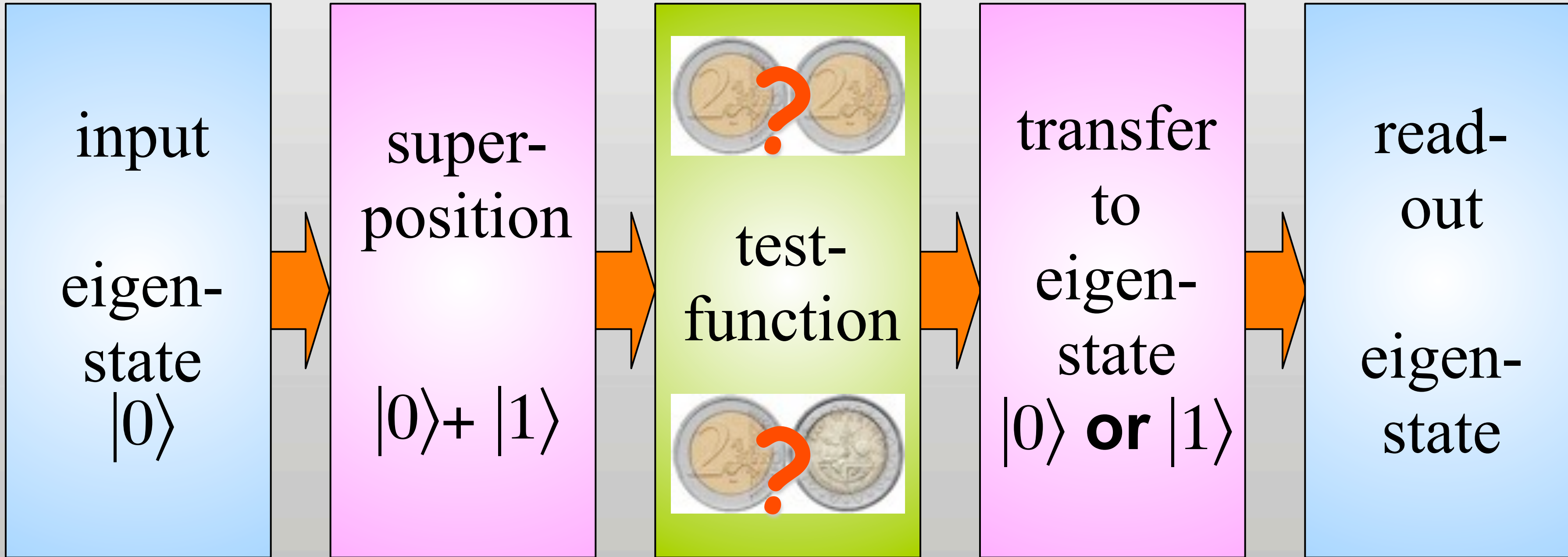
transfer to eigen-state $|0\rangle$ **or** $|1\rangle$

→

read-out

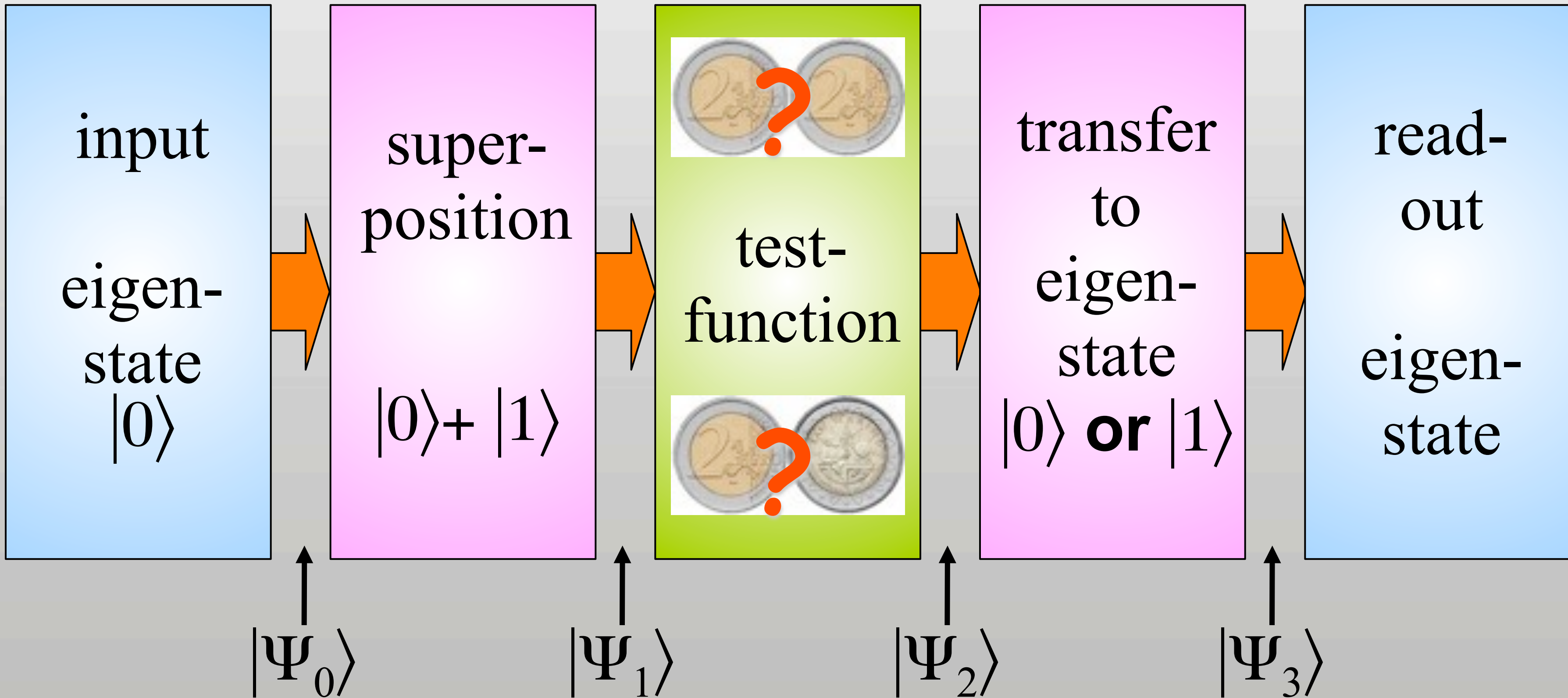
eigen-state

Deutsch algorithm: principle



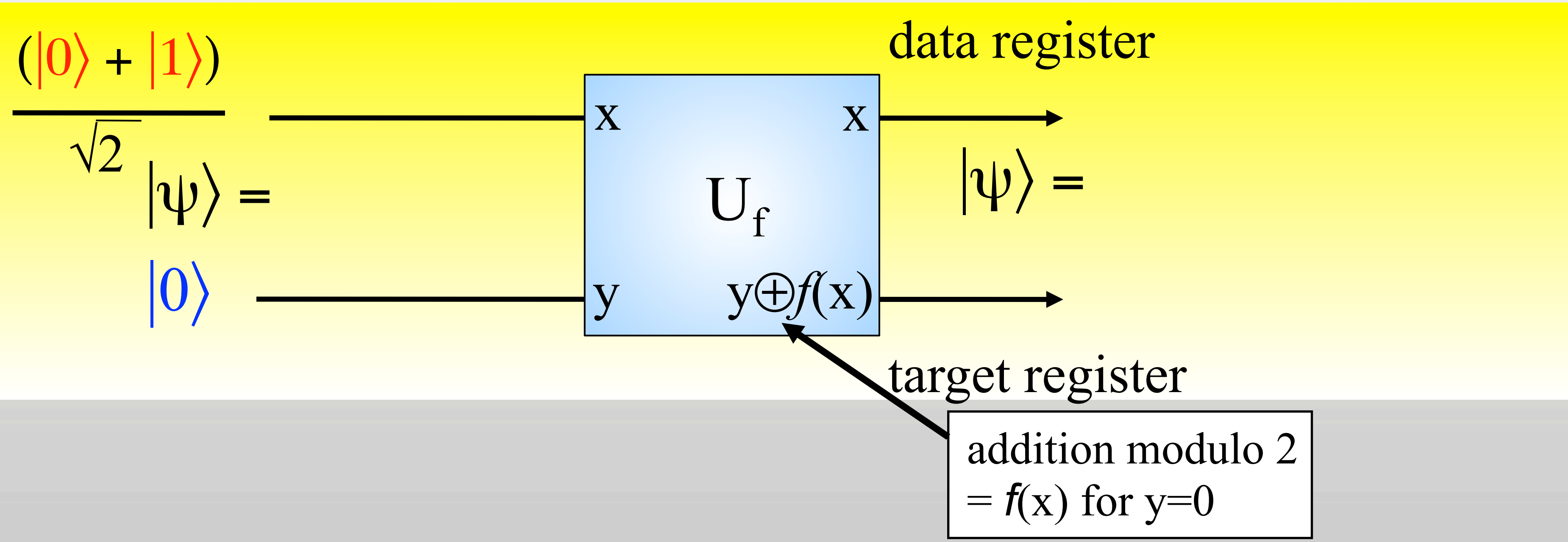
one run only!

Deutsch algorithm: principle

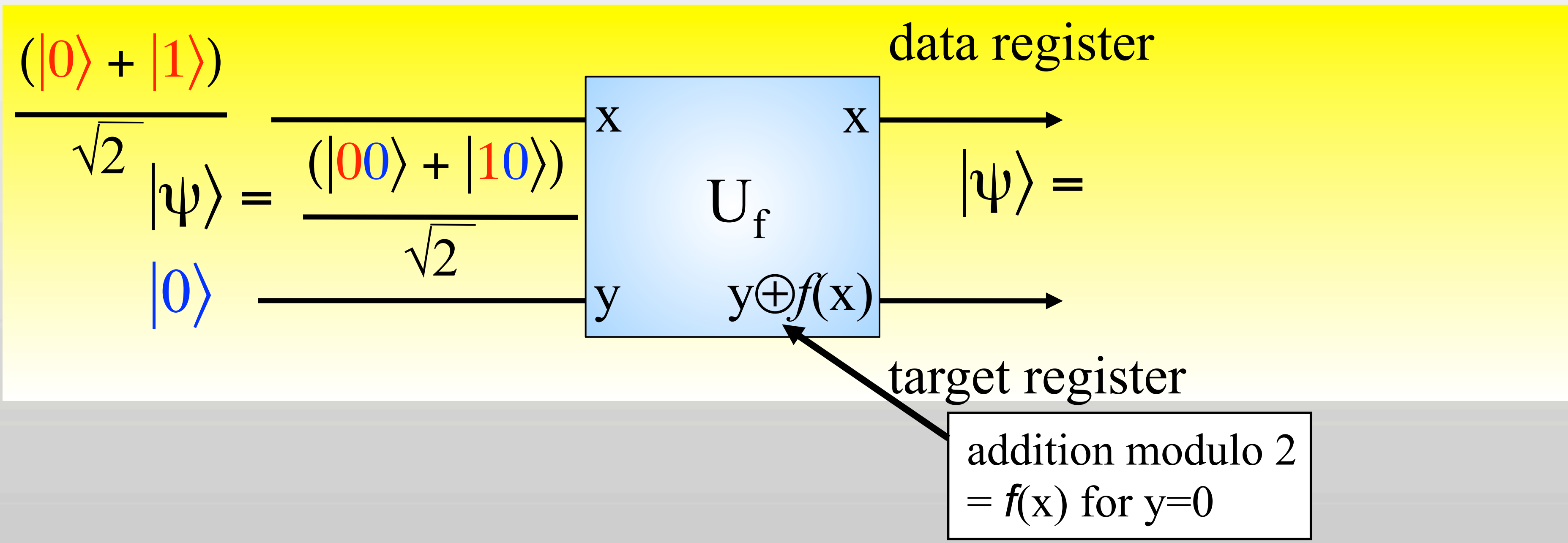


one run only!

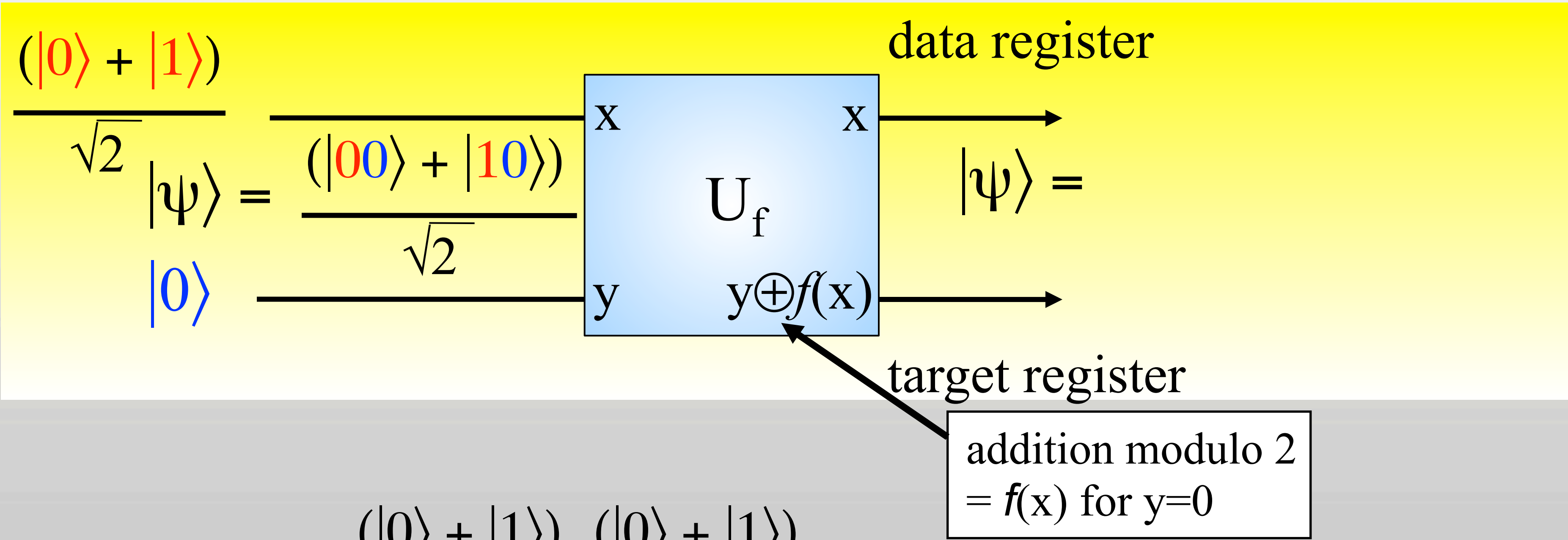
quantum circuit



quantum circuit



quantum circuit



$$H^{\otimes 2}|00\rangle = \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)$$

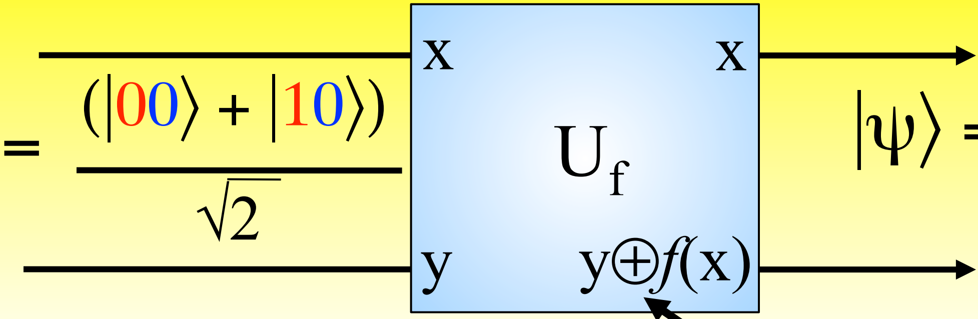
quantum circuit

data register

$$\frac{(|0\rangle + |1\rangle)}{\sqrt{2}}$$

$$|\psi\rangle = \frac{(|00\rangle + |10\rangle)}{\sqrt{2}}$$

$$|0\rangle$$



$$|\psi\rangle = \frac{(|0, f(0)\rangle + |1, f(1)\rangle)}{\sqrt{2}}$$

target register

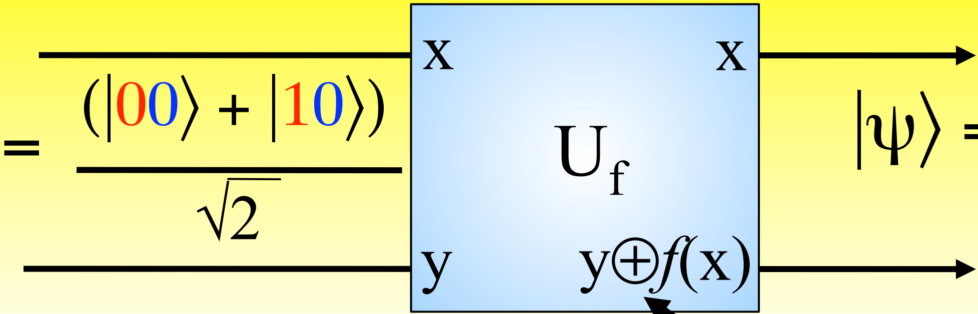
addition modulo 2
= f(x) for y=0

$$H^{\otimes 2}|00\rangle = \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)$$

quantum circuit

$$\frac{(|0\rangle + |1\rangle)}{\sqrt{2}}$$

$$|0\rangle$$



$$|\psi\rangle = \frac{(|00\rangle + |10\rangle)}{\sqrt{2}}$$

data register

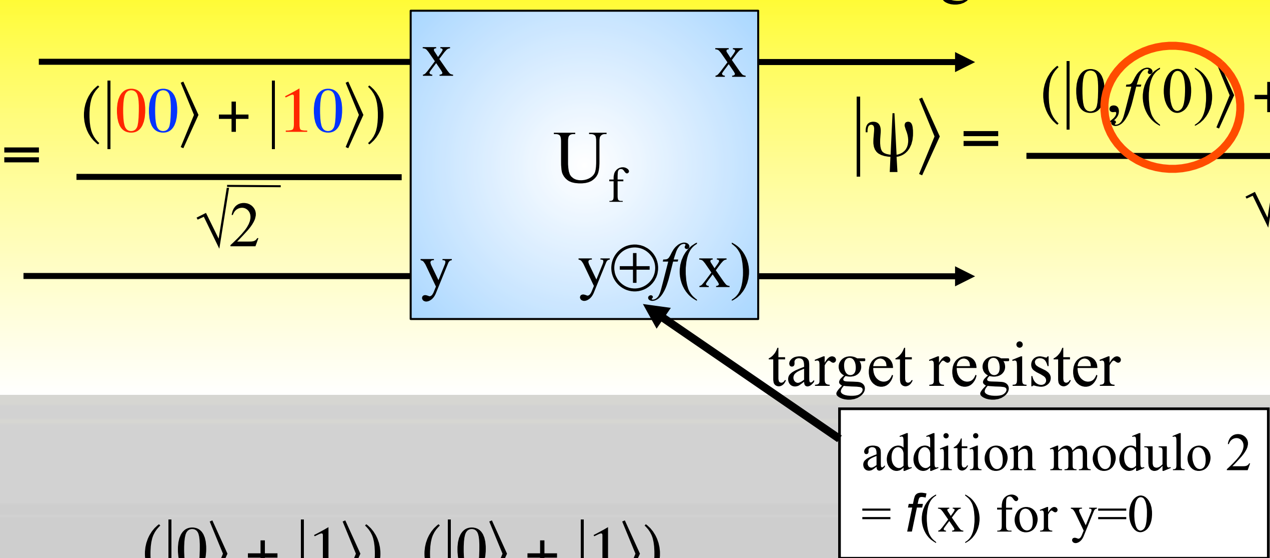
$$|\psi\rangle = \frac{(|0, f(0)\rangle + |1, f(1)\rangle)}{\sqrt{2}}$$

target register

addition modulo 2
= $f(x)$ for y=0

$$H^{\otimes 2}|00\rangle = \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)$$

quantum circuit



$$\frac{(|0\rangle + |1\rangle)}{\sqrt{2}} \quad |\psi\rangle = \frac{(|00\rangle + |10\rangle)}{\sqrt{2}} \quad |0\rangle$$

$$|\psi\rangle = \frac{(|0, f(0)\rangle + |1, f(1)\rangle)}{\sqrt{2}}$$

$$H^{\otimes 2}|00\rangle = \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)$$

evaluation of function $f(x)$ with n data qubits x and 1 target qubit

$$\frac{1}{\sqrt{2^n}} \sum_x |x\rangle f(x)$$

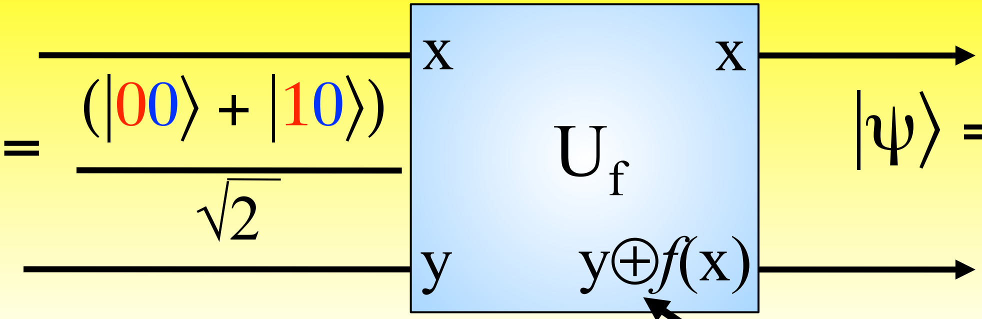
quantum circuit

$\frac{(|0\rangle + |1\rangle)}{\sqrt{2}}$

$|0\rangle$

$|\psi\rangle = \frac{(|00\rangle + |10\rangle)}{\sqrt{2}}$

data register



$|\psi\rangle = \frac{(|0,f(0)\rangle + |1,f(1)\rangle)}{\sqrt{2}}$

target register

addition modulo 2
= f(x) for y=0

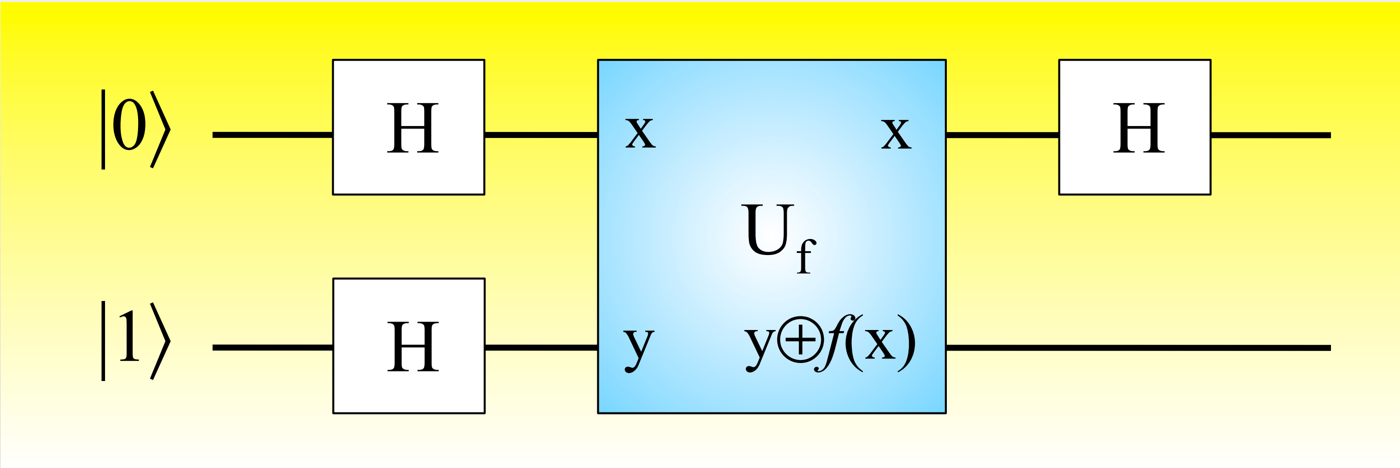
$$H^{\otimes 2}|00\rangle = \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)$$

evaluation of function f(x) with n data qubits x and 1 target qubit

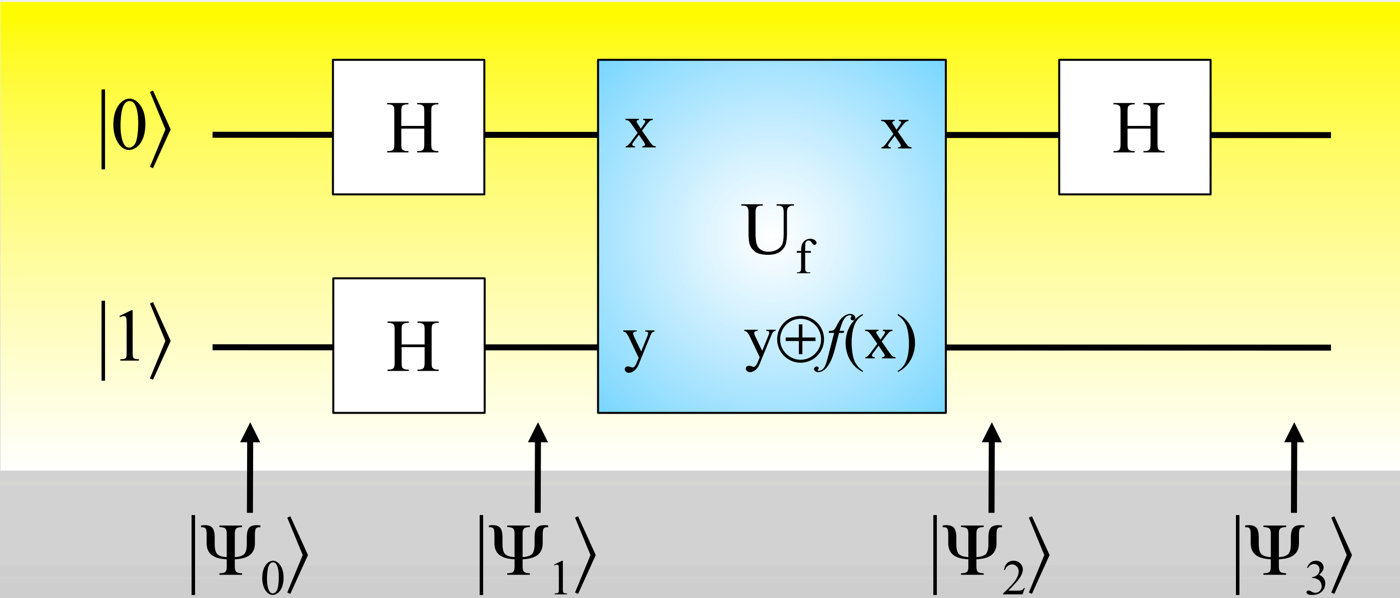
$$\frac{1}{\sqrt{2^n}} \sum_x |x\rangle f(x)$$

BUT: *How to extract information?*

Deutsch algorithm: quantum circuit

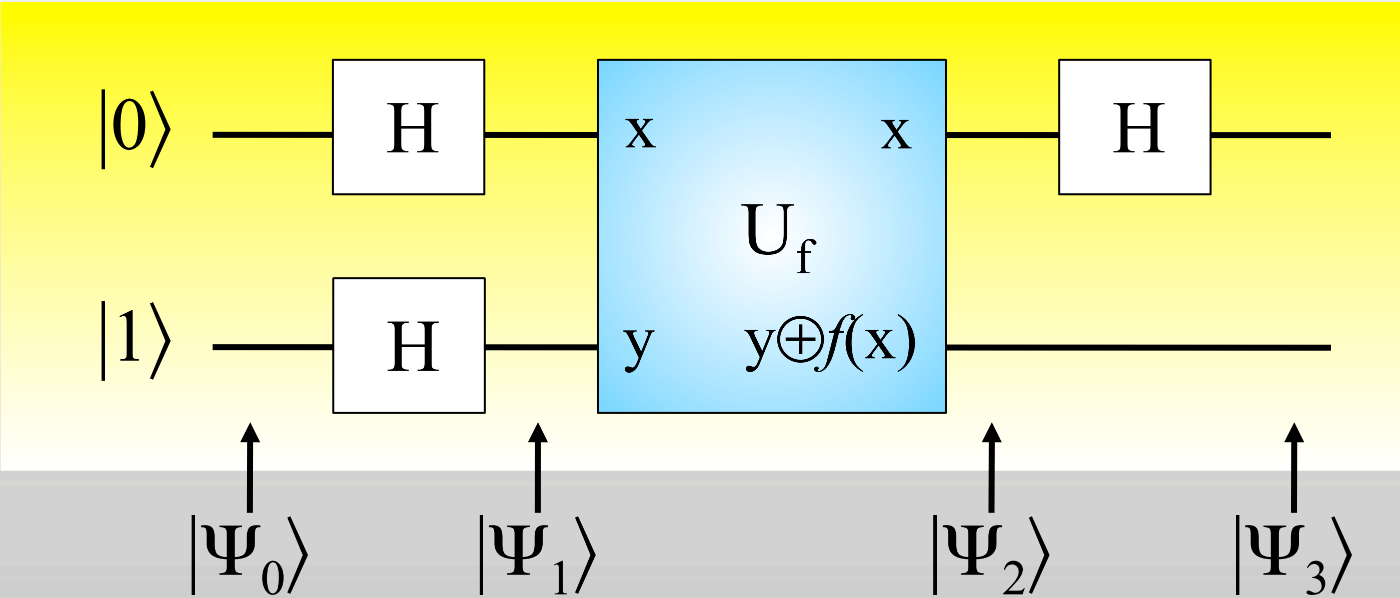


Deutsch algorithm: quantum circuit



$$|\Psi_0\rangle = |01\rangle$$

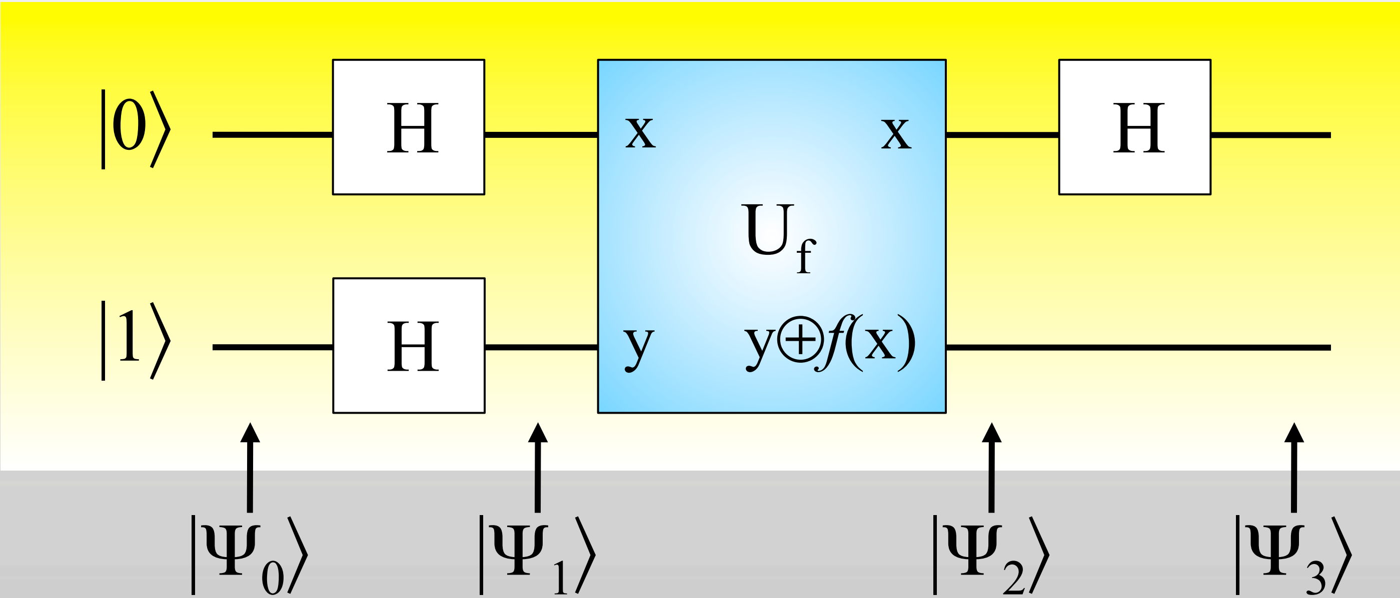
Deutsch algorithm: quantum circuit



$$|\Psi_0\rangle = |01\rangle$$

$$|\Psi_1\rangle = H^{\otimes 2}|01\rangle = \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} \frac{(|0\rangle - |1\rangle)}{\sqrt{2}} = \frac{1}{2}(|00\rangle - |01\rangle + |10\rangle - |11\rangle)$$

Deutsch algorithm: quantum circuit



$$|\Psi_0\rangle = |01\rangle$$

$$|\Psi_1\rangle = H^{\otimes 2}|01\rangle = \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} \frac{(|0\rangle - |1\rangle)}{\sqrt{2}} = \frac{1}{2}(|00\rangle - |01\rangle + |10\rangle - |11\rangle)$$

$$|\Psi_2\rangle = \mathbf{U_f}\,|\Psi_1\rangle$$

Deutsch algorithm: find U_f

$$\mathbf{U_f}\,|x,y\rangle = |x, y \oplus f(x)\rangle$$

	case 1	case 2	case 3	case 4
$f(0)$	0	1	0	1
$f(1)$	0	1	1	0
U_f				

Deutsch algorithm: find U_f

$$U_f |x,y\rangle = |x, y \oplus f(x)\rangle$$

	case 1	case 2	case 3	case 4
$f(0)$	0	1	0	1
$f(1)$	0	1	1	0
U_f				

$$U_f\text{: } |0,0\rangle \rightarrow |0,0\rangle\ ,\ |0,1\rangle \rightarrow |0,1\rangle\ ,\ |1,0\rangle \rightarrow |1,0\rangle\ ,\ |1,1\rangle \rightarrow |1,1\rangle$$

$$U_f = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Deutsch algorithm: find U_f

$$\mathbf{U_f}\,|x,y\rangle = |x, y \oplus f(x)\rangle$$

	case 1	case 2	case 3	case 4
$f(0)$	0	1	0	1
$f(1)$	0	1	1	0
U_f				

Deutsch algorithm: find U_f

$$U_f |x,y\rangle = |x, y \oplus f(x)\rangle$$

	case 1	case 2	case 3	case 4
$f(0)$	0	1	0	1
$f(1)$	0	1	1	0
U_f	ID			

Deutsch algorithm: find U_f

$$U_f |x,y\rangle = |x, y \oplus f(x)\rangle$$

	case 1	case 2	case 3	case 4
$f(0)$	0	1	0	1
$f(1)$	0	1	1	0
U_f	ID			

U_f: $|0,0\rangle \rightarrow |0,1\rangle$, $|0,1\rangle \rightarrow |0,0\rangle$, $|1,0\rangle \rightarrow |1,1\rangle$, $|1,1\rangle \rightarrow |1,0\rangle$

$$U_f = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Deutsch algorithm: find U_f

$$U_f |x,y\rangle = |x, y \oplus f(x)\rangle$$

	case 1	case 2	case 3	case 4
$f(0)$	0	1	0	1
$f(1)$	0	1	1	0
U_f	**ID**			

Deutsch algorithm: find U_f

$$U_f |x,y\rangle = |x,y \oplus f(x)\rangle$$

	case 1	case 2	case 3	case 4
$f(0)$	0	1	0	1
$f(1)$	0	1	1	0
U_f	ID	NOT		

Deutsch algorithm: find U_f

$$U_f |x,y\rangle = |x, y \oplus f(x)\rangle$$

	case 1	case 2	case 3	case 4
$f(0)$	0	1	0	1
$f(1)$	0	1	1	0
U_f	**ID**	**NOT**		

$$U_f: |0,0\rangle \rightarrow |0,0\rangle, |0,1\rangle \rightarrow |0,1\rangle, |1,0\rangle \rightarrow |1,1\rangle, |1,1\rangle \rightarrow |1,0\rangle$$

$$U_f = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Deutsch algorithm: find U_f

$$U_f |x,y\rangle = |x,y \oplus f(x)\rangle$$

	case 1	case 2	case 3	case 4
$f(0)$	0	1	0	1
$f(1)$	0	1	1	0
U_f	**ID**	**NOT**		

Deutsch algorithm: find U_f

$$U_f |x,y\rangle = |x,y \oplus f(x)\rangle$$

	case 1	case 2	case 3	case 4
$f(0)$	0	1	0	1
$f(1)$	0	1	1	0
U_f	**ID**	**NOT**	**CNOT**	

Deutsch algorithm: find U_f

$$U_f |x,y\rangle = |x, y \oplus f(x)\rangle$$

	case 1	case 2	case 3	case 4
$f(0)$	0	1	0	1
$f(1)$	0	1	1	0
U_f	**ID**	**NOT**	**CNOT**	

$$U_f: |0,0\rangle \rightarrow |0,1\rangle \,,\, |0,1\rangle \rightarrow |0,0\rangle \,,\, |1,0\rangle \rightarrow |1,0\rangle \,,\, |1,1\rangle \rightarrow |1,1\rangle$$

$$U_f = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Deutsch algorithm: find U_f

$$U_f |x,y\rangle = |x, y \oplus f(x)\rangle$$

	case 1	case 2	case 3	case 4
$f(0)$	0	1	0	1
$f(1)$	0	1	1	0
U_f	**ID**	**NOT**	**CNOT**	

Deutsch algorithm: find U_f

$$U_f \left|x,y\right\rangle = \left|x, y \oplus f(x)\right\rangle$$

	case 1	case 2	case 3	case 4
$f(0)$	0	1	0	1
$f(1)$	0	1	1	0
U_f	**ID**	**NOT**	**CNOT**	**Z-CNOT**

Deutsch algorithm: find U_f

$$\mathbf{U_f}\,|x,y\rangle = |x, y \oplus f(x)\rangle$$

	case 1	case 2	case 3	case 4
$f(0)$	0	1	0	1
$f(1)$	0	1	1	0
U_f	**ID**	**NOT**	**CNOT**	**Z-CNOT**

$$|\Psi_2\rangle = \mathbf{U_f}\,|\Psi_1\rangle = \frac{1}{2}\left(|0, f(0)\rangle - |0, 1\oplus f(0)\rangle + |1, f(1)\rangle - |1, 1\oplus f(1)\rangle\right)$$

Deutsch algorithm: find U_f

$$\mathbf{U_f}|x,y\rangle = |x,y\oplus f(x)\rangle$$

	case 1	case 2	case 3	case 4
$f(0)$	0	1	0	1
$f(1)$	0	1	1	0
U_f	**ID**	**NOT**	**CNOT**	**Z-CNOT**

false

$$|\Psi_2\rangle = \mathbf{U_f}|\Psi_1\rangle = \frac{1}{2}\left(|0, f(0)\rangle - |0,1\oplus f(0)\rangle + |1, f(1)\rangle - |1,1\oplus f(1)\rangle\right)$$

$$|\Psi_2\rangle = \mathbf{U_f}|\Psi_1\rangle = \frac{1}{2}(|0\rangle + |1\rangle)\,(|f(0)\rangle - |1\oplus f(0)\rangle)$$

Deutsch algorithm: find U_f

$$U_f |x,y\rangle = |x, y \oplus f(x)\rangle$$

	case 1	case 2	case 3	case 4
$f(0)$	0	1	0	1
$f(1)$	0	1	1	0
U_f	**ID**	**NOT**	**CNOT**	**Z-CNOT**

false (cases 1–2) | true (cases 3–4)

$$|\Psi_2\rangle = U_f |\Psi_1\rangle = \frac{1}{2}(|0, f(0)\rangle - |0, 1\oplus f(0)\rangle + |1, f(1)\rangle - |1, 1\oplus f(1)\rangle)$$

$$|\Psi_2\rangle = U_f |\Psi_1\rangle = \frac{1}{2}(|0\rangle + |1\rangle)\,(|f(0)\rangle - |1\oplus f(0)\rangle)$$

$$|\Psi_2\rangle = U_f |\Psi_1\rangle = \frac{1}{2}(|0\rangle - |1\rangle)\,(|f(0)\rangle - |1\oplus f(0)\rangle)$$

Deutsch algorithm: find U_f

$$U_f |x,y\rangle = |x, y \oplus f(x)\rangle$$

	case 1	case 2	case 3	case 4
$f(0)$	0	1	0	1
$f(1)$	0	1	1	0
U_f	ID	NOT	CNOT	Z-CNOT

Cases 1 and 2: **false** — Cases 3 and 4: **true**

$$|\Psi_2\rangle = U_f |\Psi_1\rangle = \frac{1}{2}(|0, f(0)\rangle - |0, 1\oplus f(0)\rangle + |1, f(1)\rangle - |1, 1\oplus f(1)\rangle)$$

$$|\Psi_2\rangle = U_f |\Psi_1\rangle = \frac{1}{2}(|0\rangle + |1\rangle)(|f(0)\rangle - |1\oplus f(0)\rangle)$$

$$|\Psi_2\rangle = U_f |\Psi_1\rangle = \frac{1}{2}(|0\rangle - |1\rangle)(|f(0)\rangle - |1\oplus f(0)\rangle)$$

information encoded in phase of x-qubit

Deutsch algorithm: get the answer

$$|\Psi_3\rangle = \mathbf{H}_x \, |\Psi_2\rangle$$

Deutsch algorithm: get the answer

$$|\Psi_3\rangle = \mathbf{H}_x |\Psi_2\rangle$$

false : $\mathbf{H}|\Psi_2\rangle_x = \mathbf{H}\frac{1}{2}(|0\rangle + |1\rangle) = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\frac{1}{2}\begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$$= \frac{1}{\sqrt{2}}|0\rangle = \frac{1}{\sqrt{2}}\,|f(0) \oplus f(1)\rangle$$

Deutsch algorithm: get the answer

$$|\Psi_3\rangle = \mathbf{H}_x\,|\Psi_2\rangle$$

false : $\mathbf{H}|\Psi_2\rangle_x = \mathbf{H}\,\frac{1}{2}(|0\rangle + |1\rangle) = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\frac{1}{2}\begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$$= \frac{1}{\sqrt{2}}|0\rangle = \frac{1}{\sqrt{2}}\,|f(0) \oplus f(1)\rangle$$

true : $\mathbf{H}|\Psi_2\rangle_x = \mathbf{H}\,\frac{1}{2}(|0\rangle - |1\rangle) = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\frac{1}{2}\begin{bmatrix} 1 \\ -1 \end{bmatrix}$

$$= \frac{1}{\sqrt{2}}|1\rangle = \frac{1}{\sqrt{2}}\,|f(0) \oplus f(1)\rangle$$

Deutsch algorithm: get the answer

$$|\Psi_3\rangle = \mathbf{H}_x\,|\Psi_2\rangle = \left| f(0) \oplus f(1) \right\rangle \left(\frac{|f(0)\rangle - |1 \oplus f(0)\rangle}{\sqrt{2}} \right)$$

read-out

false: $\mathbf{H}|\Psi_2\rangle_x = \mathbf{H}\,\frac{1}{2}(|0\rangle + |1\rangle) = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \frac{1}{2}\begin{pmatrix} 1 \\ 1 \end{pmatrix}$

$$= \frac{1}{\sqrt{2}}|0\rangle = \frac{1}{\sqrt{2}}\left| f(0) \oplus f(1) \right\rangle$$

true: $\mathbf{H}|\Psi_2\rangle_x = \mathbf{H}\,\frac{1}{2}(|0\rangle - |1\rangle) = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \frac{1}{2}\begin{pmatrix} 1 \\ -1 \end{pmatrix}$

$$= \frac{1}{\sqrt{2}}|1\rangle = \frac{1}{\sqrt{2}}\left| f(0) \oplus f(1) \right\rangle$$

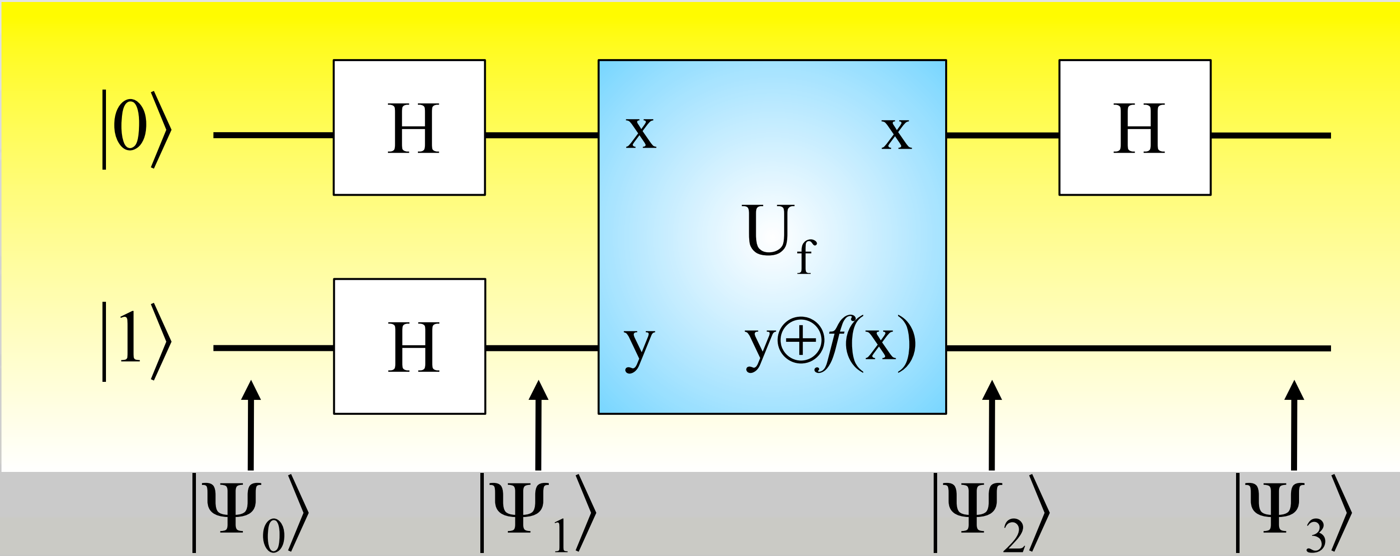
Deutsch algorithm: summary

- evaluates a global property of a function f(x) with a single run

Deutsch algorithm: summary

- evaluates a global property of a function f(x) with a single run

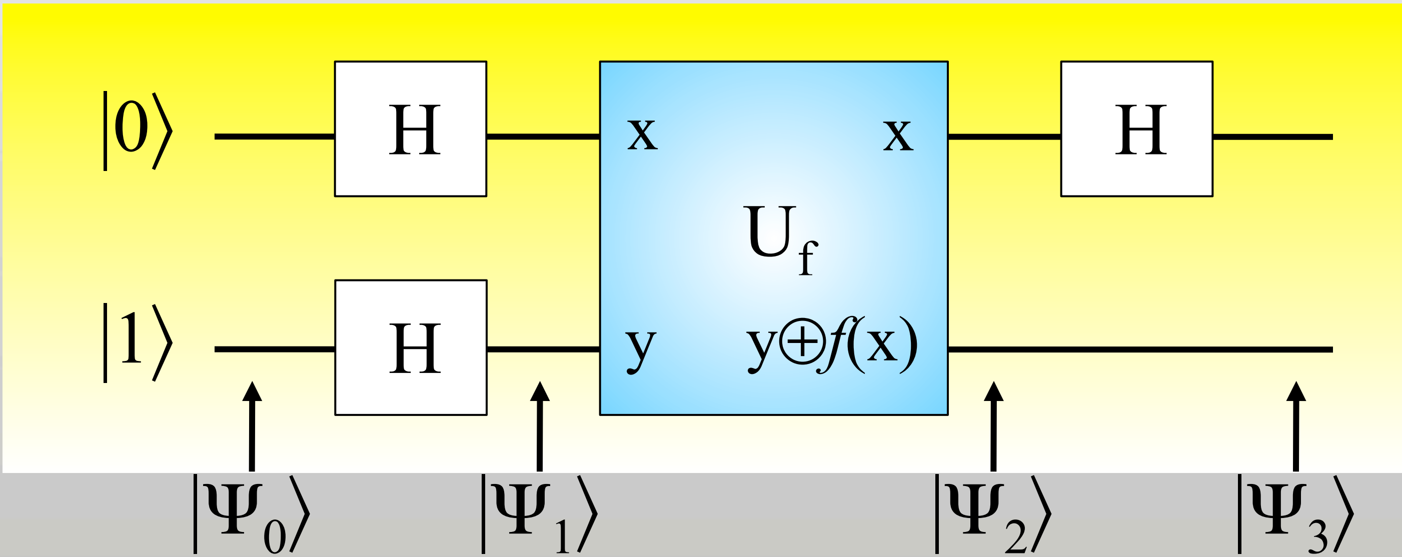
Example: f(x) → f(0) = 1, f(1) = 1



Deutsch algorithm: summary

- evaluates a global property of a function f(x) with a single run

Example: f(x) → f(0) = 1, f(1) = 1

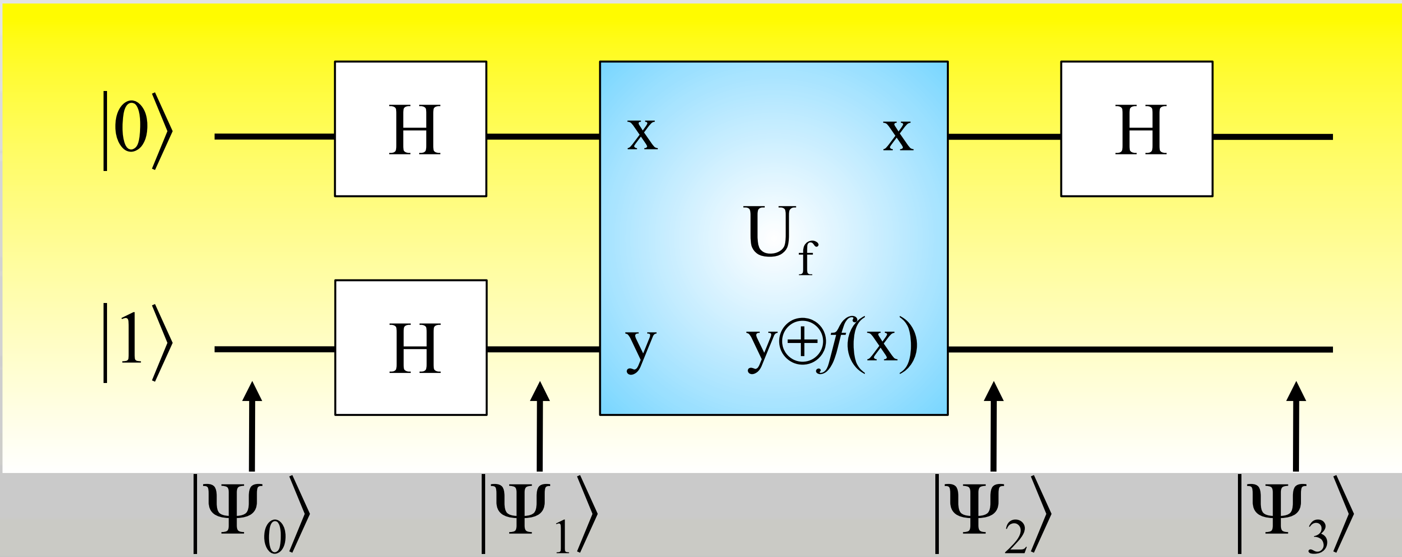


$$|\Psi_0\rangle = |01\rangle$$

Deutsch algorithm: summary

- evaluates a global property of a function f(x) with a single run

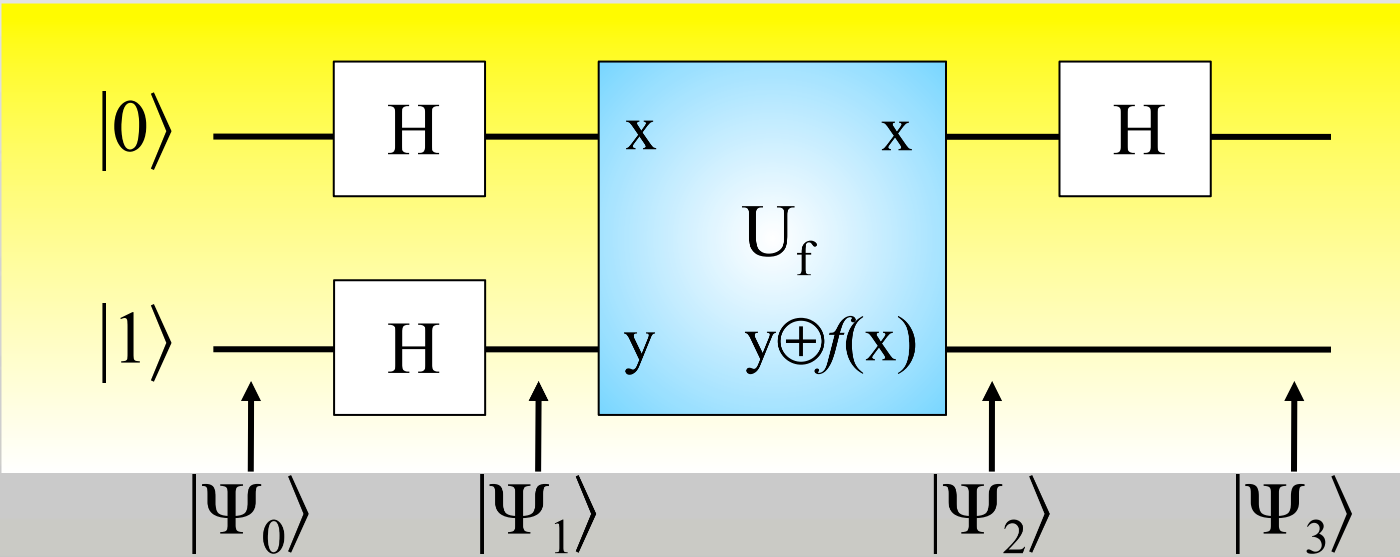
Example: f(x) → f(0) = 1, f(1) = 1



Deutsch algorithm: summary

- evaluates a global property of a function f(x) with a single run

Example: $f(x) \rightarrow f(0) = 1,\ f(1) = 1$

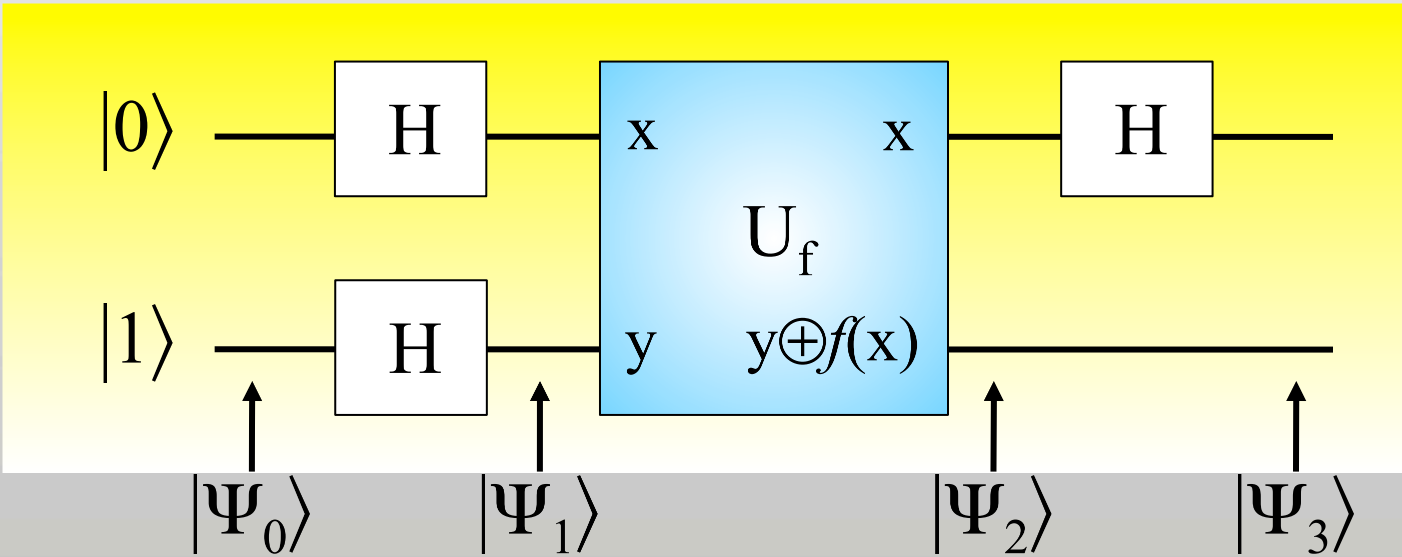


$$|\Psi_1\rangle = H^{\otimes 2}|01\rangle = \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} \frac{(|0\rangle - |1\rangle)}{\sqrt{2}} = \frac{1}{2}(|00\rangle - |01\rangle + |10\rangle - |11\rangle)$$

Deutsch algorithm: summary

- evaluates a global property of a function f(x) with a single run

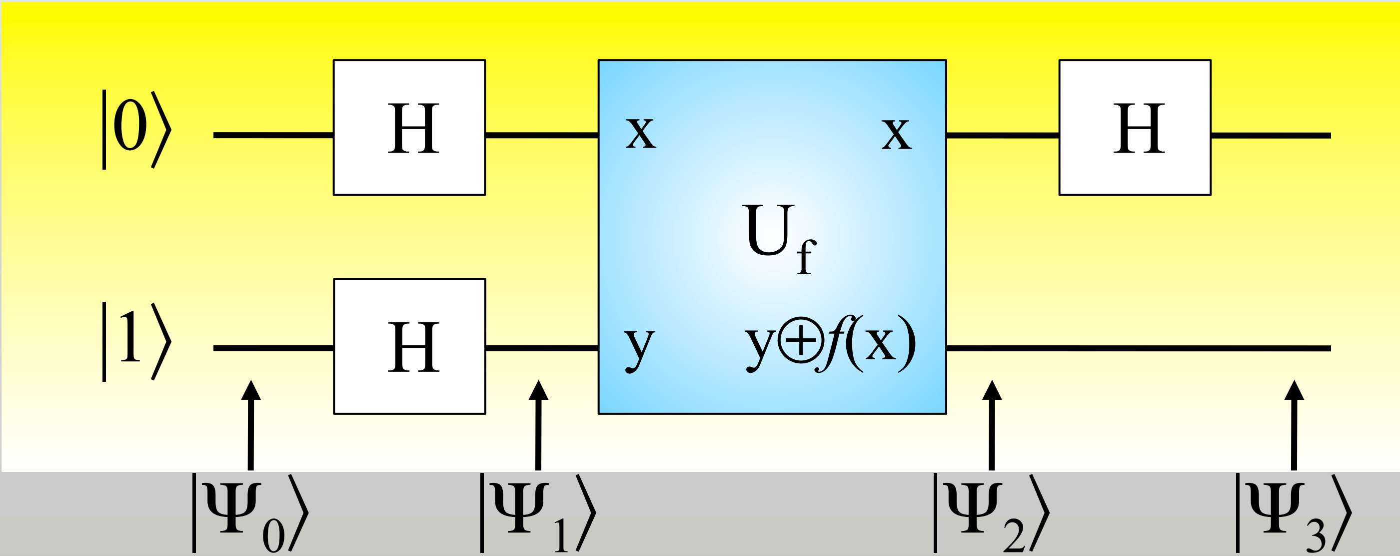
Example: f(x) → f(0) = 1, f(1) = 1



Deutsch algorithm: summary

- evaluates a global property of a function f(x) with a single run

Example: f(x) → f(0) = 1, f(1) = 1

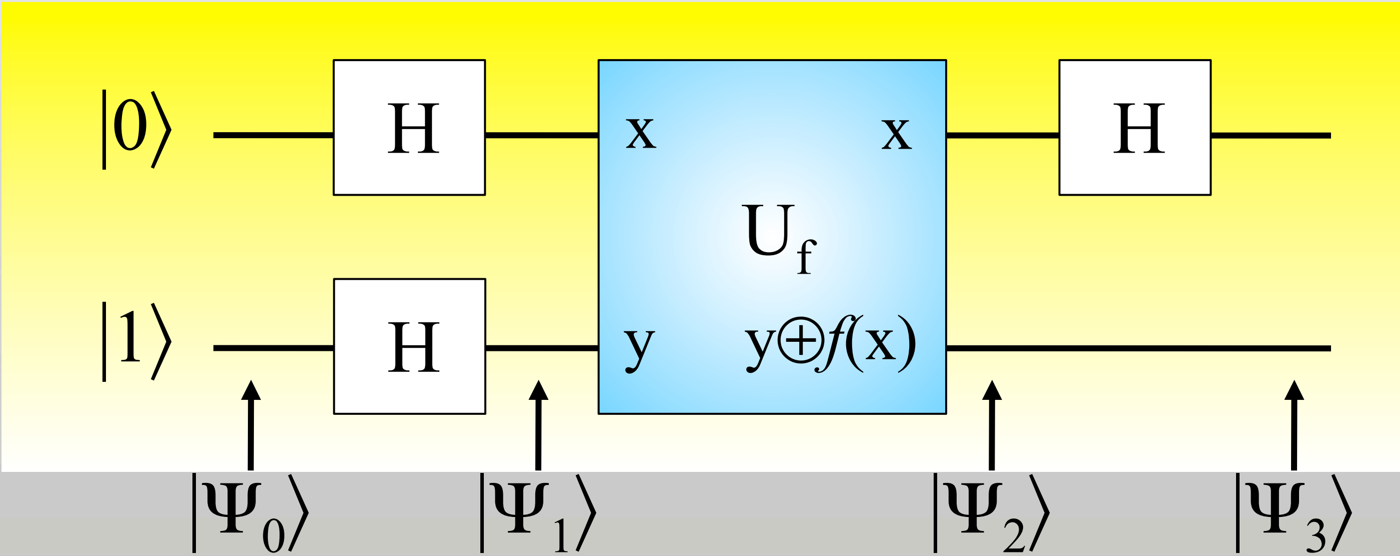


$$|\Psi_2\rangle = \mathbf{U}_f|\Psi_1\rangle = \frac{1}{2}\begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}\begin{pmatrix} 1 \\ -1 \\ 1 \\ -1 \end{pmatrix} = \frac{1}{2}\begin{pmatrix} -1 \\ 1 \\ -1 \\ 1 \end{pmatrix} = \frac{(-|00\rangle + |01\rangle - |10\rangle + |11\rangle)}{2}$$

Deutsch algorithm: summary

- evaluates a global property of a function f(x) with a single run

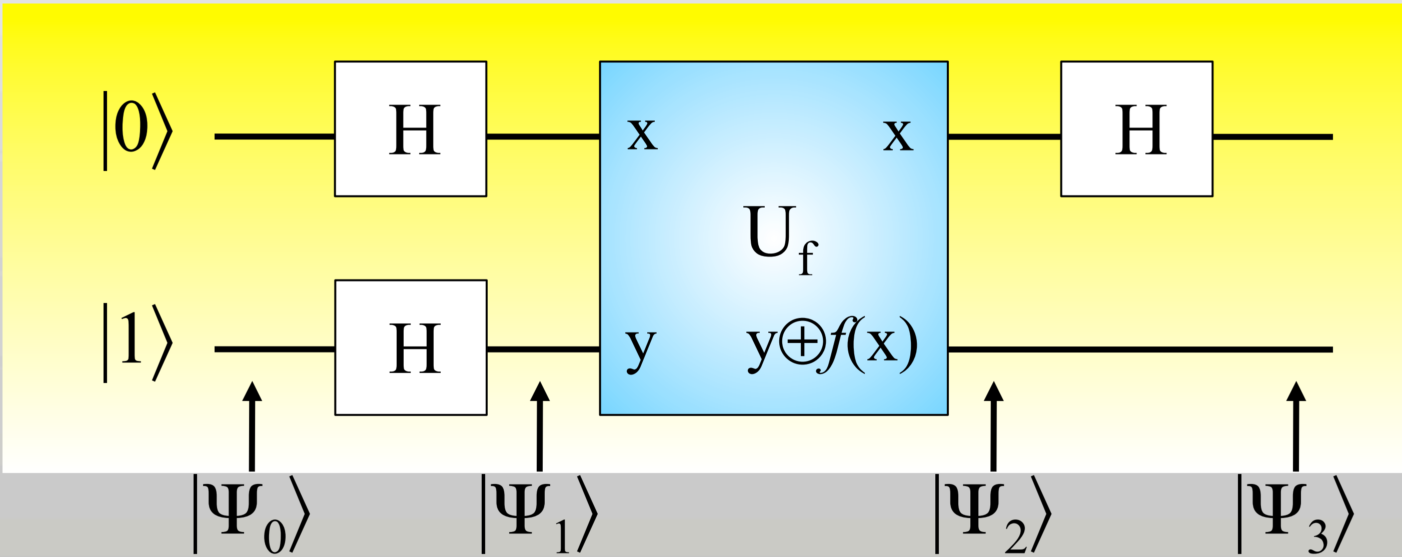
Example: $f(x) \rightarrow f(0) = 1,\ f(1) = 1$



Deutsch algorithm: summary

- evaluates a global property of a function f(x) with a single run

Example: f(x) → f(0) = 1, f(1) = 1

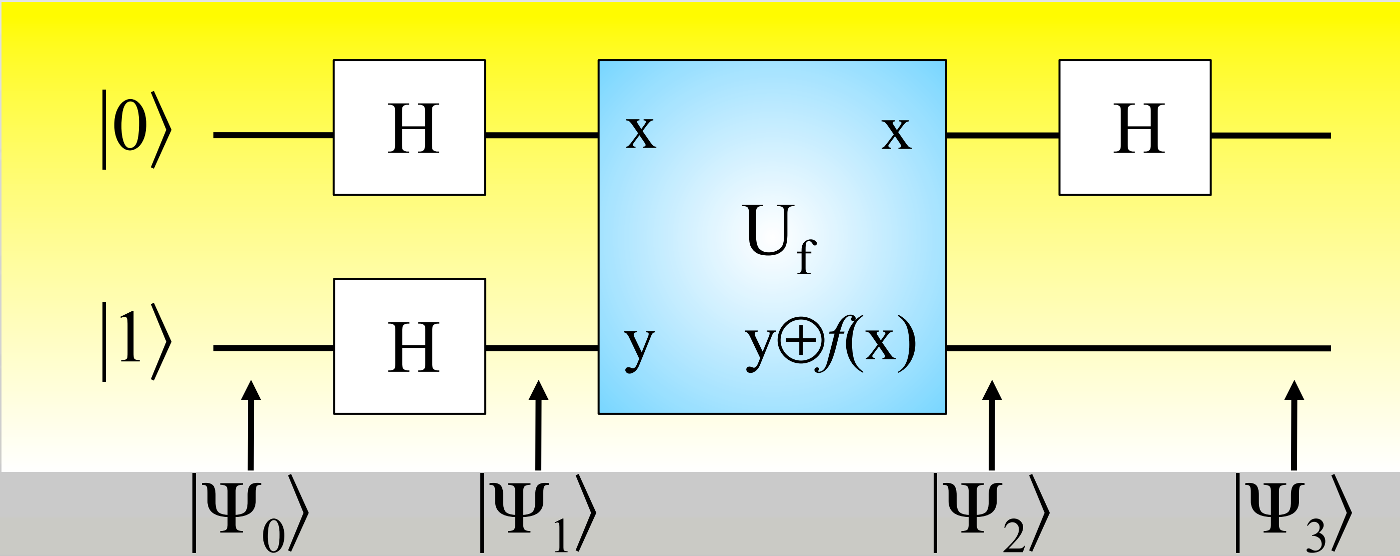


$$|\Psi_2\rangle = \frac{(-|00\rangle + |01\rangle - |10\rangle + |11\rangle)}{2} = \frac{1}{2}(|0\rangle + |1\rangle)(-|0\rangle + |1\rangle)$$

Deutsch algorithm: summary

- evaluates a global property of a function f(x) with a single run

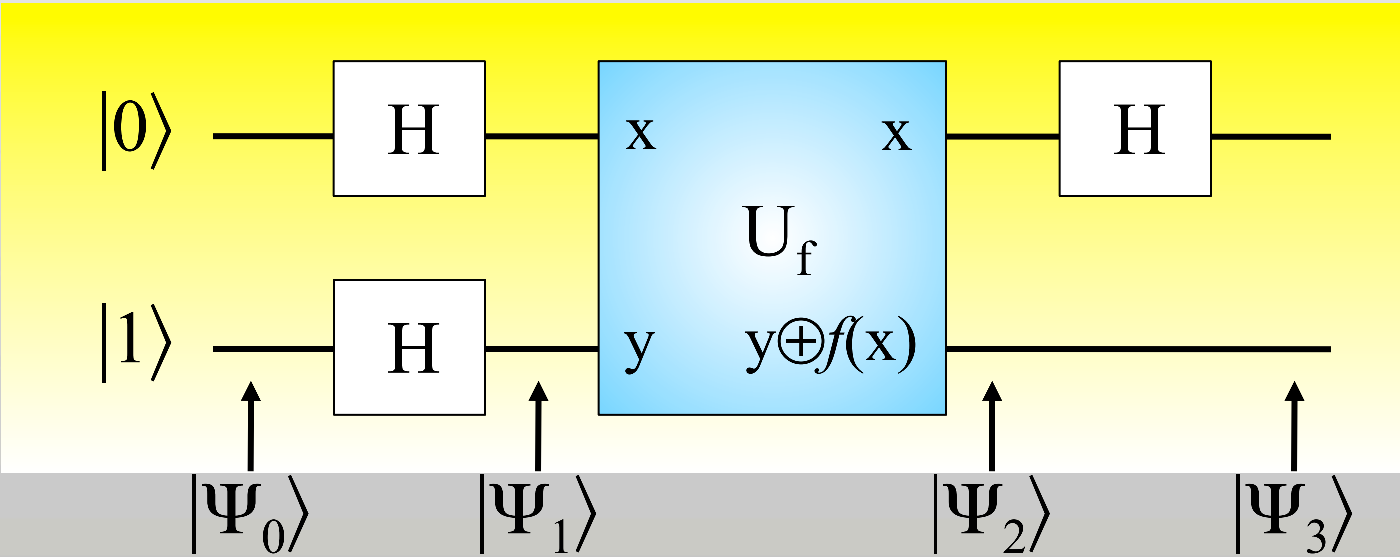
Example: $f(x) \rightarrow f(0) = 1,\ f(1) = 1$



Deutsch algorithm: summary

- evaluates a global property of a function f(x) with a single run

Example: f(x) → f(0) = 1, f(1) = 1



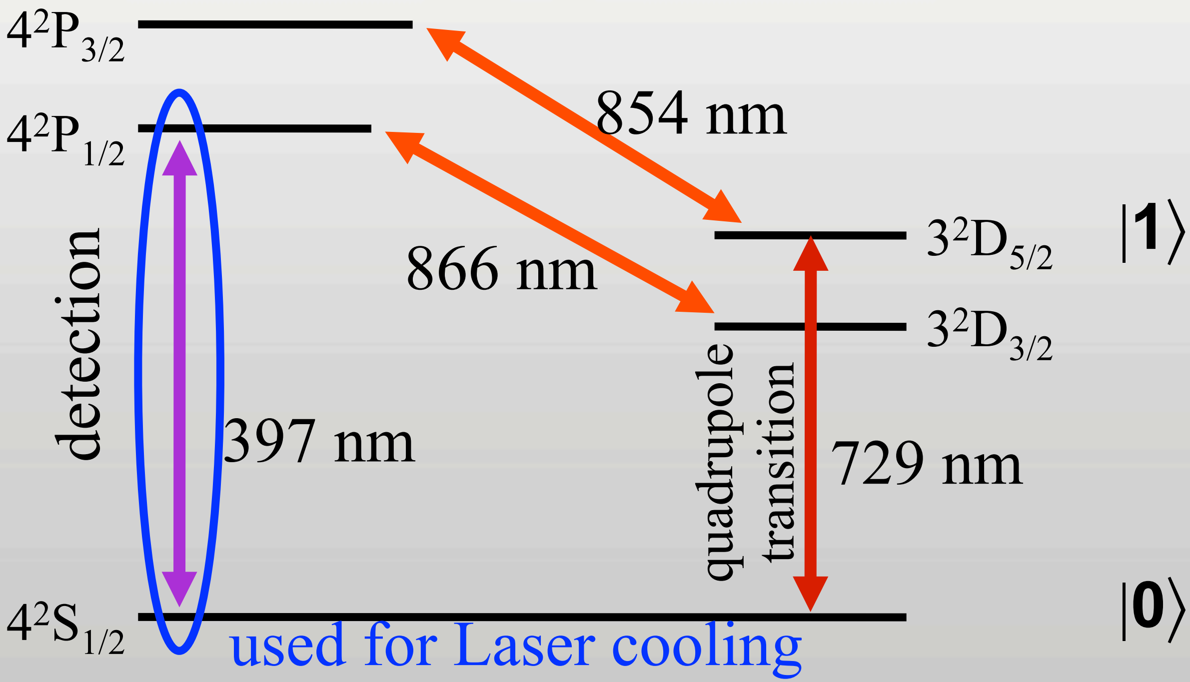
$$\mathbf{H}|\Psi_2\rangle_x = \mathbf{H}\,\frac{1}{2}(|0\rangle + |1\rangle) = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}\frac{1}{2}\begin{pmatrix}1\\ 1\end{pmatrix}$$

$$= \frac{1}{\sqrt{2}}|0\rangle$$

Deutsch algorithm: implementation

Roos et al: Phys. Rev. Lett. 83, 4713 (1999)

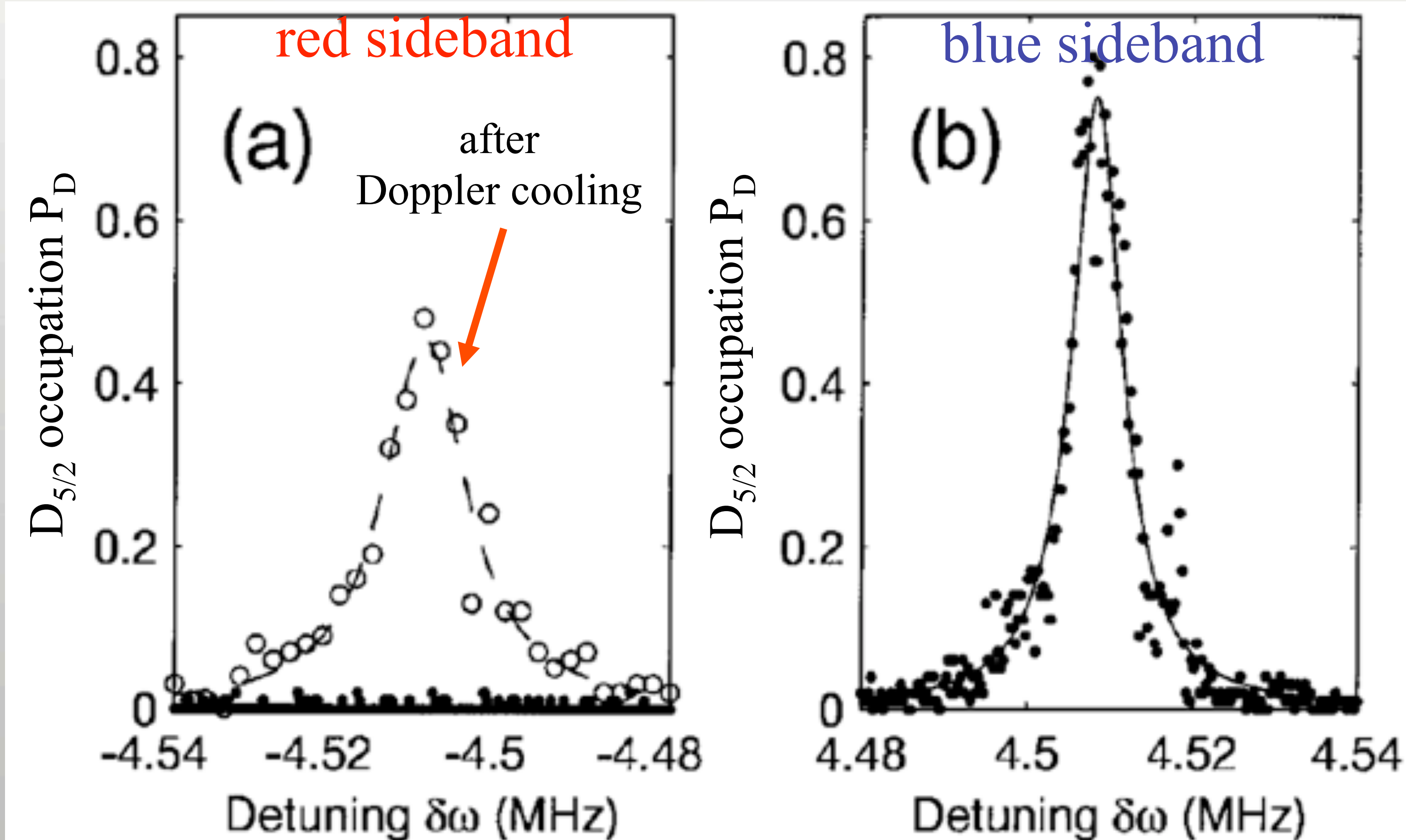
$^{40}Ca^+$



Deutsch algorithm: implementation

Roos et al: Phys. Rev. Lett. 83, 4713 (1999)

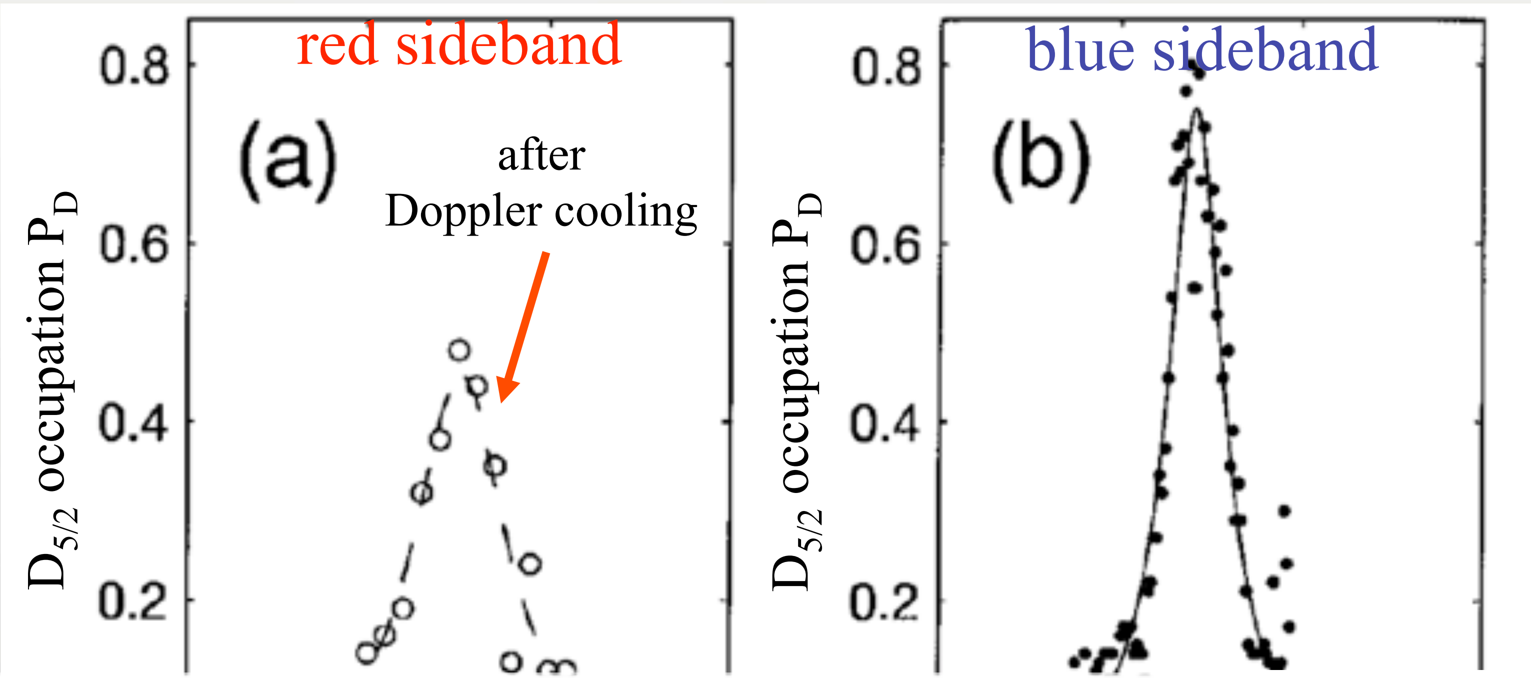
$^{40}Ca^+$

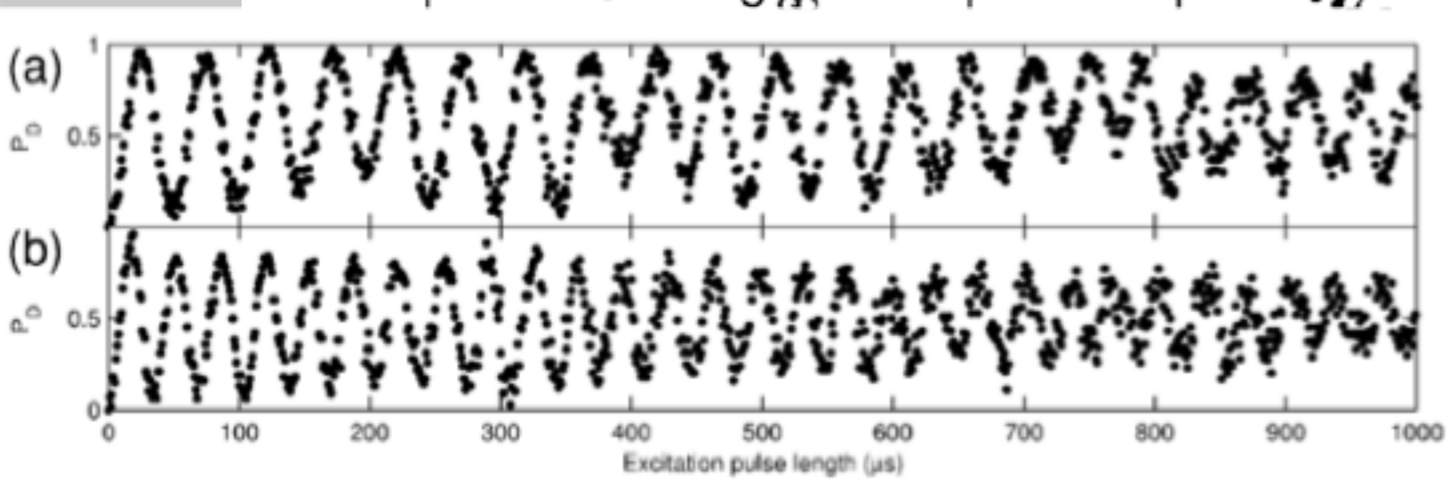


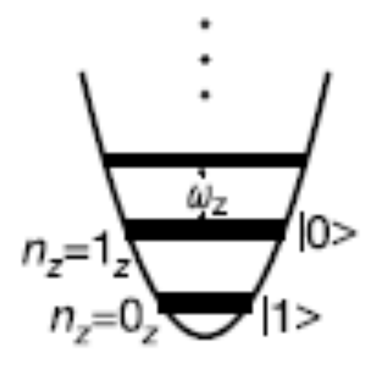
Deutsch algorithm: implementation

Roos et al: Phys. Rev. Lett. 83, 4713 (1999)

$^{40}Ca^+$



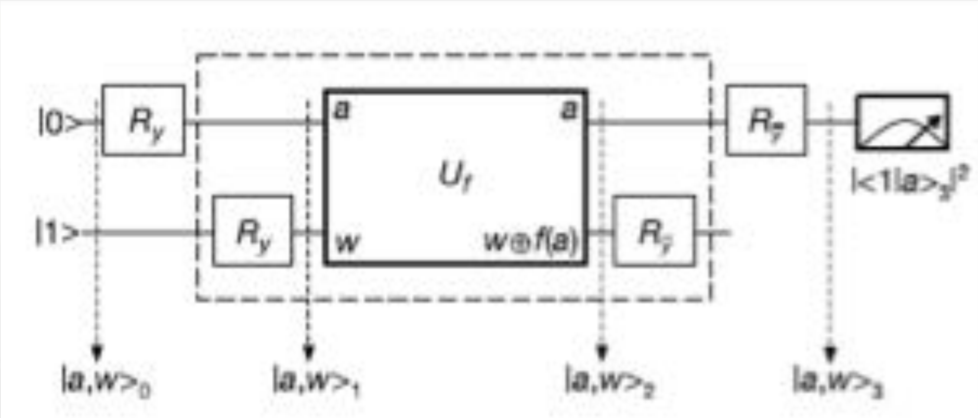




11

Deutsch algorithm: implementation

Gulde et al: Nature 421, 48 (2003)



Deutsch algorithm: implementation

Gulde et al: Nature 421, 48 (2003)

<u>case 1:</u> U_f = ID

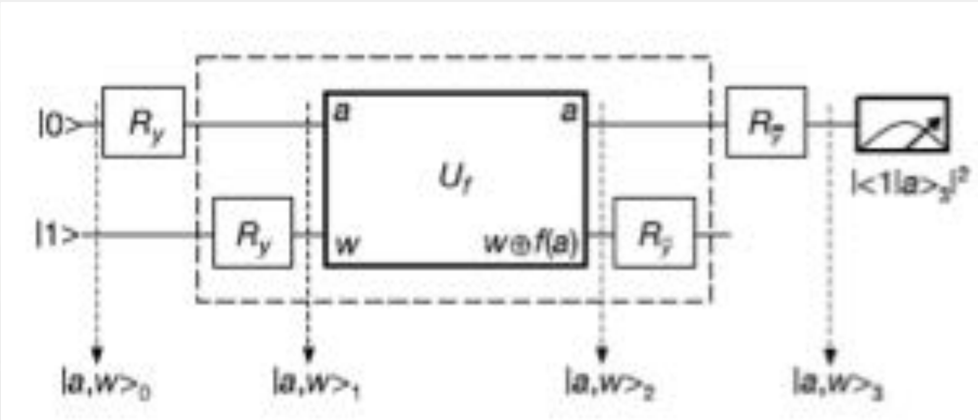


Table 3 **Implementations of $R_{\bar{y}-}, U_f, R_{y-}$**

	Logic	Laser pulses
f_1	$R_{\bar{y}-}R_{y-}$	No pulses

Deutsch algorithm: implementation

Gulde et al: Nature 421, 48 (2003)

<u>case 1:</u> U_f = ID

<u>case 3:</u> U_f = CNOT

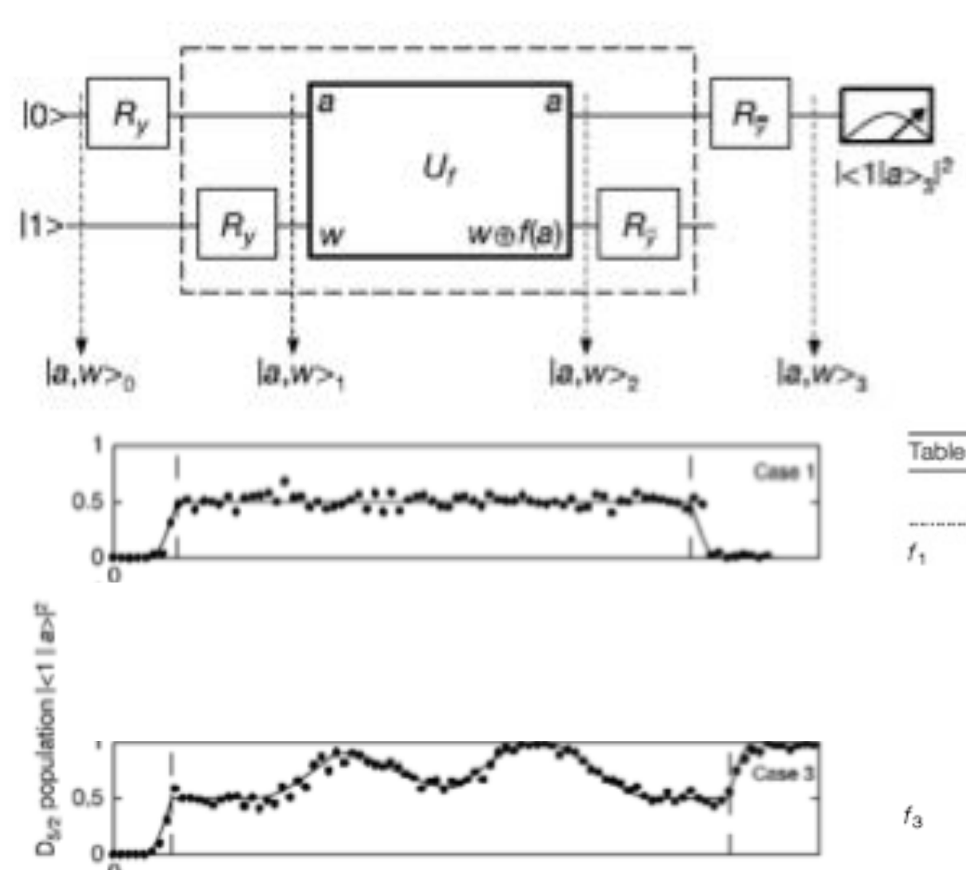
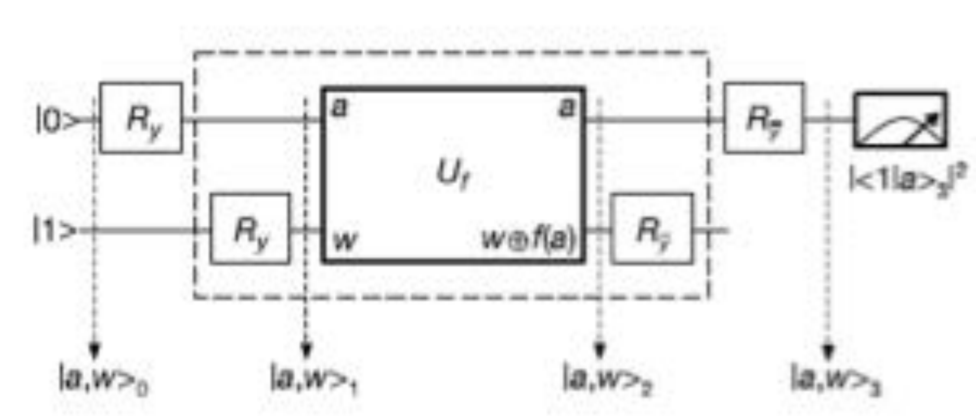


Table 3 **Implementations of $R_{\bar{y}_w}, U_f, R_{y_w}$**

	Logic	Laser pulses
f_1	$R_{\bar{y}_w} R_{y_w}$	No pulses
f_3	$R_{\bar{y}_w}$ CNOT R_{y_w}	$R^+\left(\frac{\pi}{\sqrt{2}},0\right)R^+\left(\pi,\frac{\pi}{2}\right)R^+\left(\frac{\pi}{\sqrt{2}},0\right)R^+\left(\pi,\frac{\pi}{2}\right)$

Deutsch algorithm: implementation

Gulde et al: Nature 421, 48 (2003)



case 1: U_f = ID

case 3: U_f = CNOT

case 4: U_f = Z-CNOT

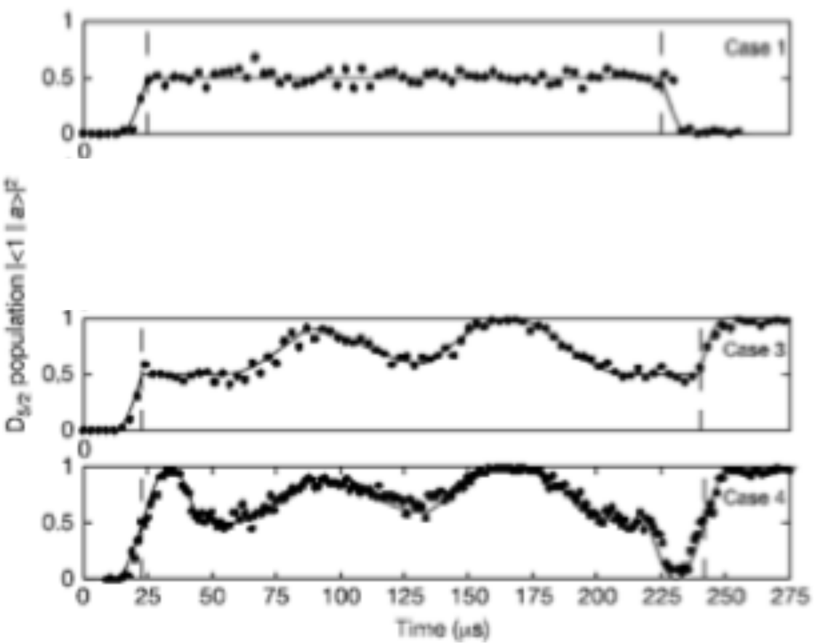
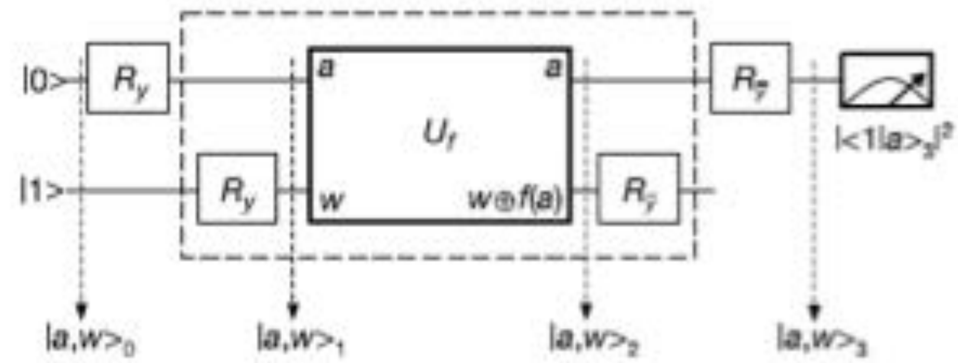


Table 3 **Implementations of $R_{\bar{y}_w}, U_f, R_{y_w}$**

	Logic	Laser pulses
f_1	$R_{\bar{y}_w} R_{y_w}$	No pulses
f_3	$R_{\bar{y}_w}$ CNOT R_{y_w}	$R^+\left(\frac{\pi}{\sqrt{2}},0\right)R^+\left(\pi,\frac{\pi}{2}\right)R^+\left(\frac{\pi}{\sqrt{2}},0\right)R^+\left(\pi,\frac{\pi}{2}\right)$
f_4	$R_{\bar{y}_w}$ Z-CNOT R_{y_w}	$R(\pi,0)R^+\left(\frac{\pi}{\sqrt{2}},0\right)R^+\left(\pi,\frac{\pi}{2}\right)R^+\left(\frac{\pi}{\sqrt{2}},0\right)R^+\left(\pi,\frac{\pi}{2}\right)R(\pi,0)$

Deutsch algorithm: implementation

Gulde et al: Nature 421, 48 (2003)



case 1: U_f = ID

case 2: U_f = NOT

case 3: U_f = CNOT

case 4: U_f = Z-CNOT

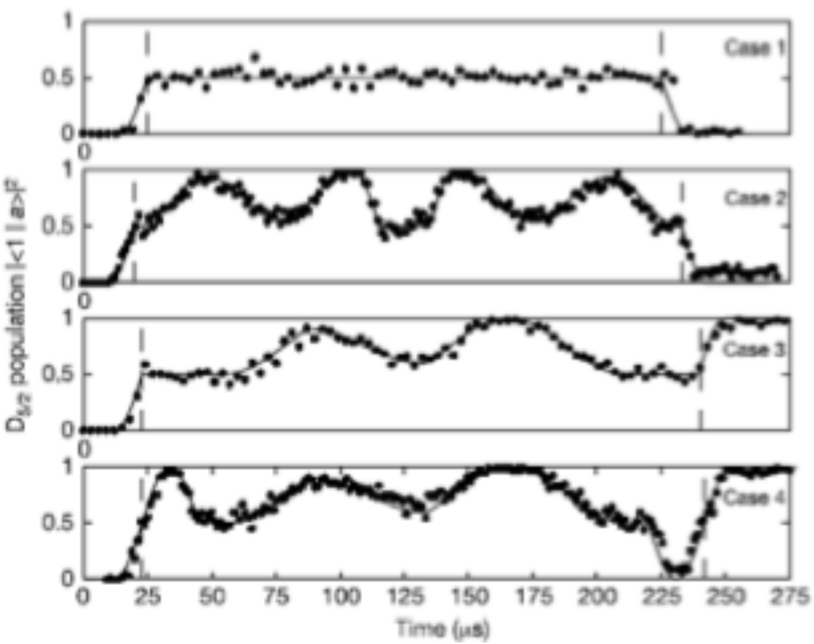


Table 3 **Implementations of $R_{\bar{y}_w} U_{f_a} R_{y_w}$**

	Logic	Laser pulses
f_1	$R_{\bar{y}_w} R_{y_w}$	No pulses
f_2	$R_{\bar{y}_w}$ SWAP^{-1} NOT$_a$ SWAP R_{y_w}	$R^+\left(\frac{\pi}{\sqrt{2}},0\right)R^+\left(\frac{2\pi}{\sqrt{2}},\varphi_{SWAP}\right)R^+\left(\frac{\pi}{\sqrt{2}},0\right)$ $R(\frac{\pi}{2},0)R(\pi,\frac{\pi}{2})R(\frac{\pi}{2},\pi)$ $R^+\left(\frac{\pi}{\sqrt{2}},\pi\right)R^+\left(\frac{2\pi}{\sqrt{2}},\pi+\varphi_{SWAP}\right)R^+\left(\frac{\pi}{\sqrt{2}},\pi\right)$
f_3	$R_{\bar{y}_w}$ CNOT R_{y_w}	$R^+\left(\frac{\pi}{\sqrt{2}},0\right)R^+(\pi,\frac{\pi}{2})R^+\left(\frac{\pi}{\sqrt{2}},0\right)R^+(\pi,\frac{\pi}{2})$
f_4	$R_{\bar{y}_w}$ Z-CNOT R_{y_w}	$R(\pi,0)R^+\left(\frac{\pi}{\sqrt{2}},0\right)R^+(\pi,\frac{\pi}{2})R^+\left(\frac{\pi}{\sqrt{2}},0\right)R^+(\pi,\frac{\pi}{2})R(\pi,0)$

Deutsch algorithm: Fidelity

Gulde et al: Nature 421, 48 (2003)

Table 2 **Expected and measured results of the complete Deutsch–Jozsa algorithm**

	Constant		Balanced	
	Case 1	Case 2	Case 3	Case 4
Expected $\lvert\langle 1 \mid a\rangle\rvert^2$	0	0	1	1
Measured $\lvert\langle 1 \mid a\rangle\rvert^2$	0.019(6)	0.087(6)	0.975(4)	0.975(2)
Expected $\lvert\langle 1 \mid w\rangle\rvert^2$	1	1	1	1
Measured $\lvert\langle 1 \mid w\rangle\rvert^2$	–	0.90(1)	0.931(9)	0.986(4)

Table 3 **Implementations of $R_{\bar{y}_w} U_{f_n} R_{y_w}$**

	Logic	Laser pulses
f_1	$R_{\bar{y}_w} R_{y_w}$	No pulses
f_2	$R_{\bar{y}_w}$ SWAP^{-1} NOT$_a$ SWAP R_{y_w}	$R^+\left(\frac{\pi}{\sqrt{2}},0\right)R^+\left(\frac{2\pi}{\sqrt{2}},\varphi_{SWAP}\right)R^+\left(\frac{\pi}{\sqrt{2}},0\right)$
		$R(\frac{\pi}{2},0)R(\pi,\frac{\pi}{2})R(\frac{\pi}{2},\pi)$
		$R^+\left(\frac{\pi}{\sqrt{2}},\pi\right)R^+\left(\frac{2\pi}{\sqrt{2}},\pi+\varphi_{SWAP}\right)R^+\left(\frac{\pi}{\sqrt{2}},\pi\right)$
f_3	$R_{\bar{y}_w}$ CNOT R_{y_w}	$R^+\left(\frac{\pi}{\sqrt{2}},0\right)R^+(\pi,\frac{\pi}{2})R^+\left(\frac{\pi}{\sqrt{2}},0\right)R^+(\pi,\frac{\pi}{2})$
f_4	$R_{\bar{y}_w}$ Z-CNOT R_{y_w}	$R(\pi,0)R^+\left(\frac{\pi}{\sqrt{2}},0\right)R^+(\pi,\frac{\pi}{2})R^+\left(\frac{\pi}{\sqrt{2}},0\right)R^+(\pi,\frac{\pi}{2})R(\pi,0)$

Needle in a haystack

Grover algorithm: search in an unsorted database

Lov Grover,
Bell labs

"Quantum mechanics helps in searching for a needle in a haystack"

Phys. Rev. Lett. **79**, 325 (1997)

Search in a database

Example: search for a specific number in a phonebook

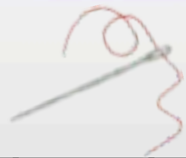
<u>We need:</u>

- $N = 2^n$ entries with index $x = 0 \ldots N-1$.
- A “detector” function f(x):

  - Entry x is no solution: $f(x) = 0$
  - Entry x is a solution: $f(x) = 1$

0	0	0	0	0	1	0	0

The oracle



0	0	0	0	0	1	0	0

Grover's algorithm minimizes calls to "oracle"

Classical: on average N/2 calls to oracle.

Quantum: number of calls $\propto \sqrt{N}$.

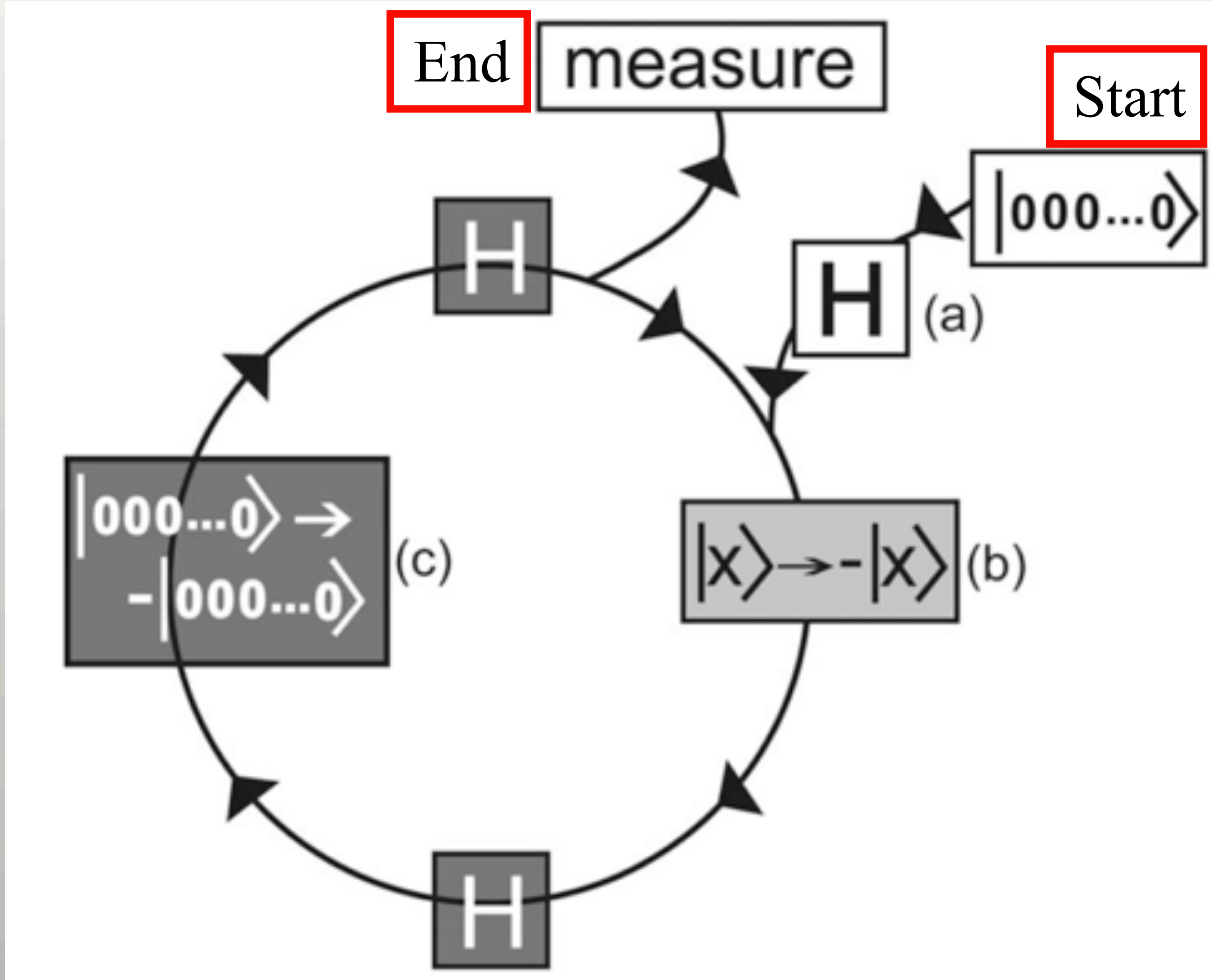
0	1	0	0
$\|00\rangle$	$\|01\rangle$	$\|10\rangle$	$\|11\rangle$

f(x)=1 if entrance is solution

$|x\rangle$: addresses of data register

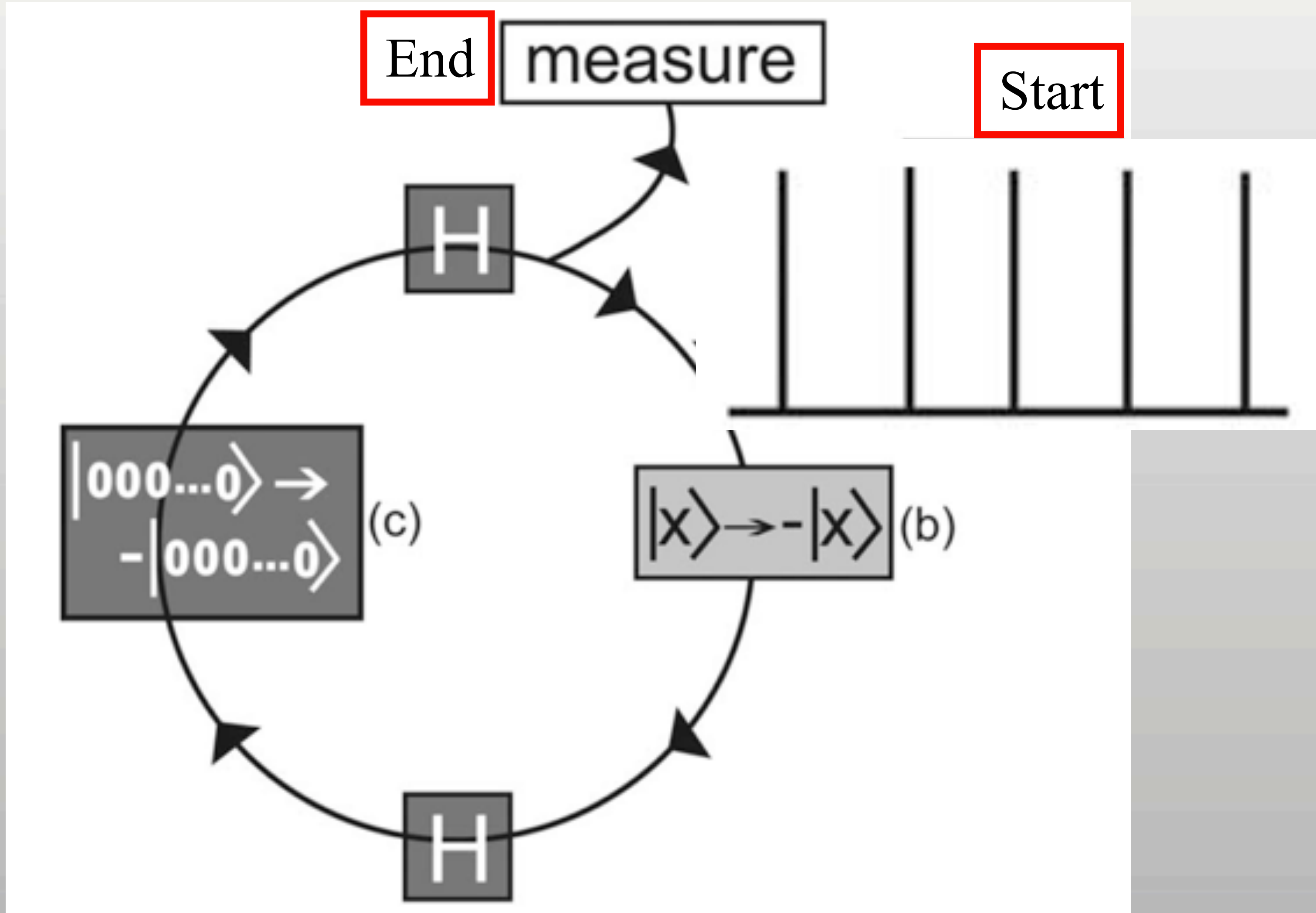
depicting Grover's algorithm

Brickmann et al: Phys. Rev. A. 72, 050306(R) (2005)



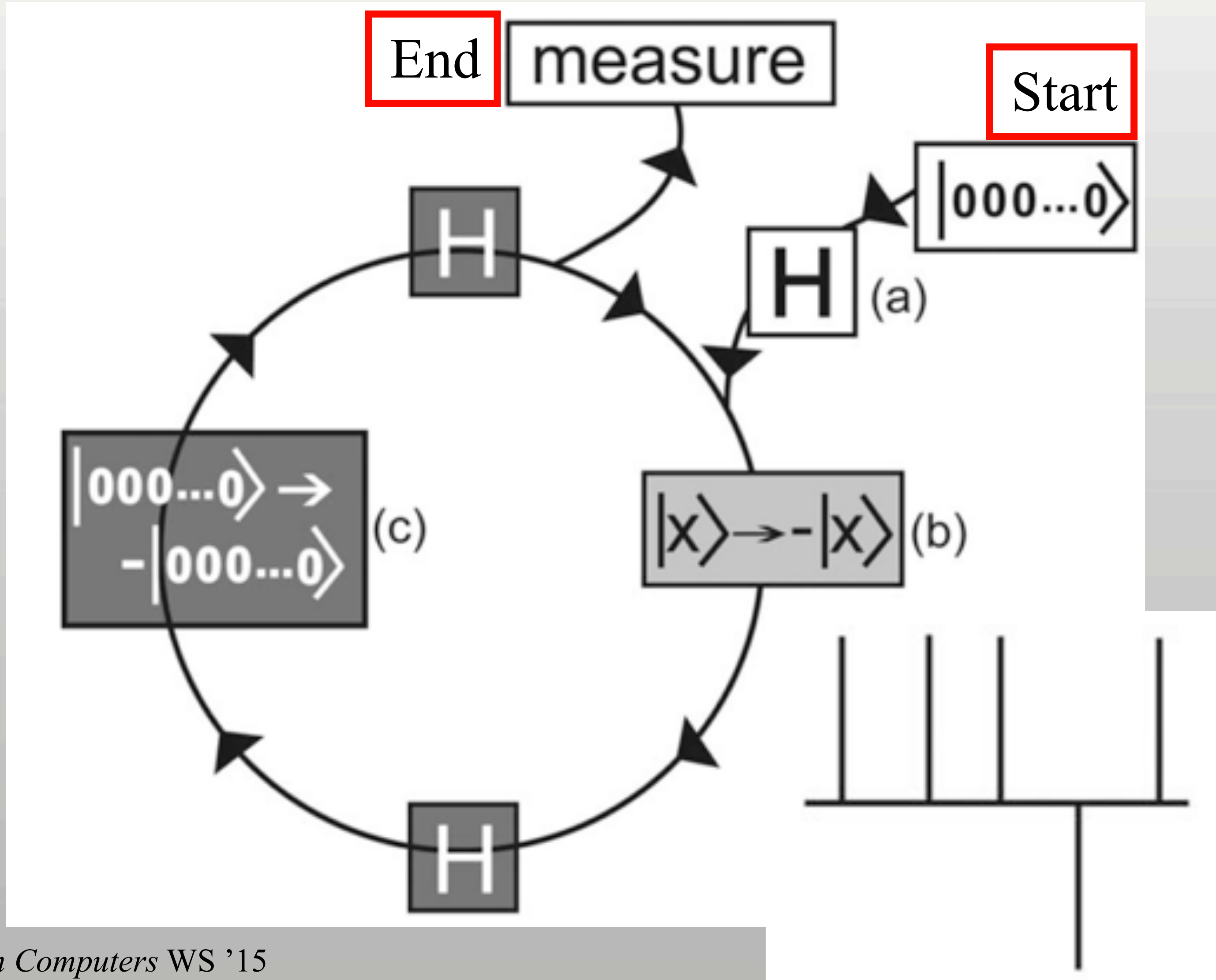
depicting Grover's algorithm

Brickmann et al: Phys. Rev. A. 72, 050306(R) (2005)



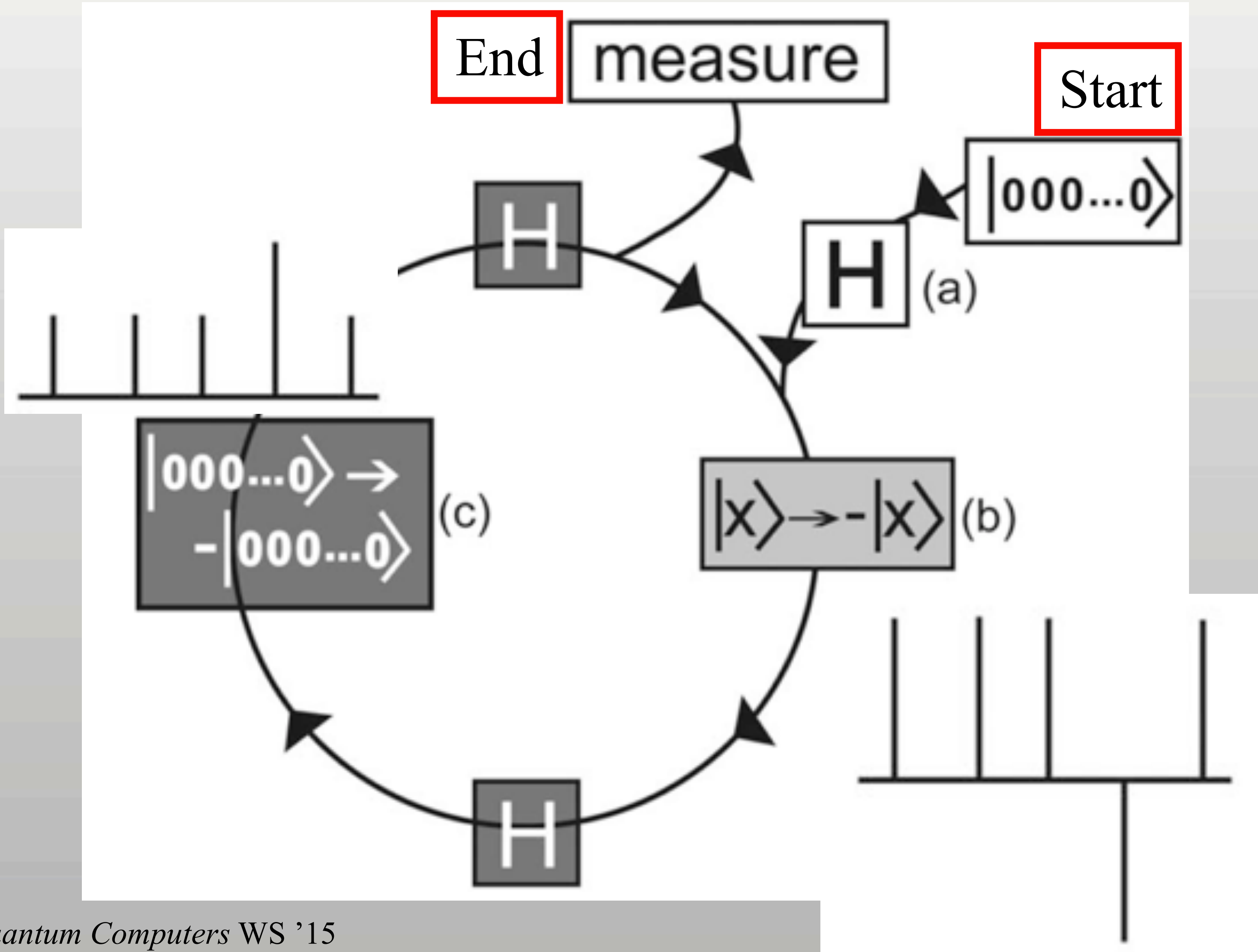
depicting Grover's algorithm

Brickmann et al: Phys. Rev. A. 72, 050306(R) (2005)



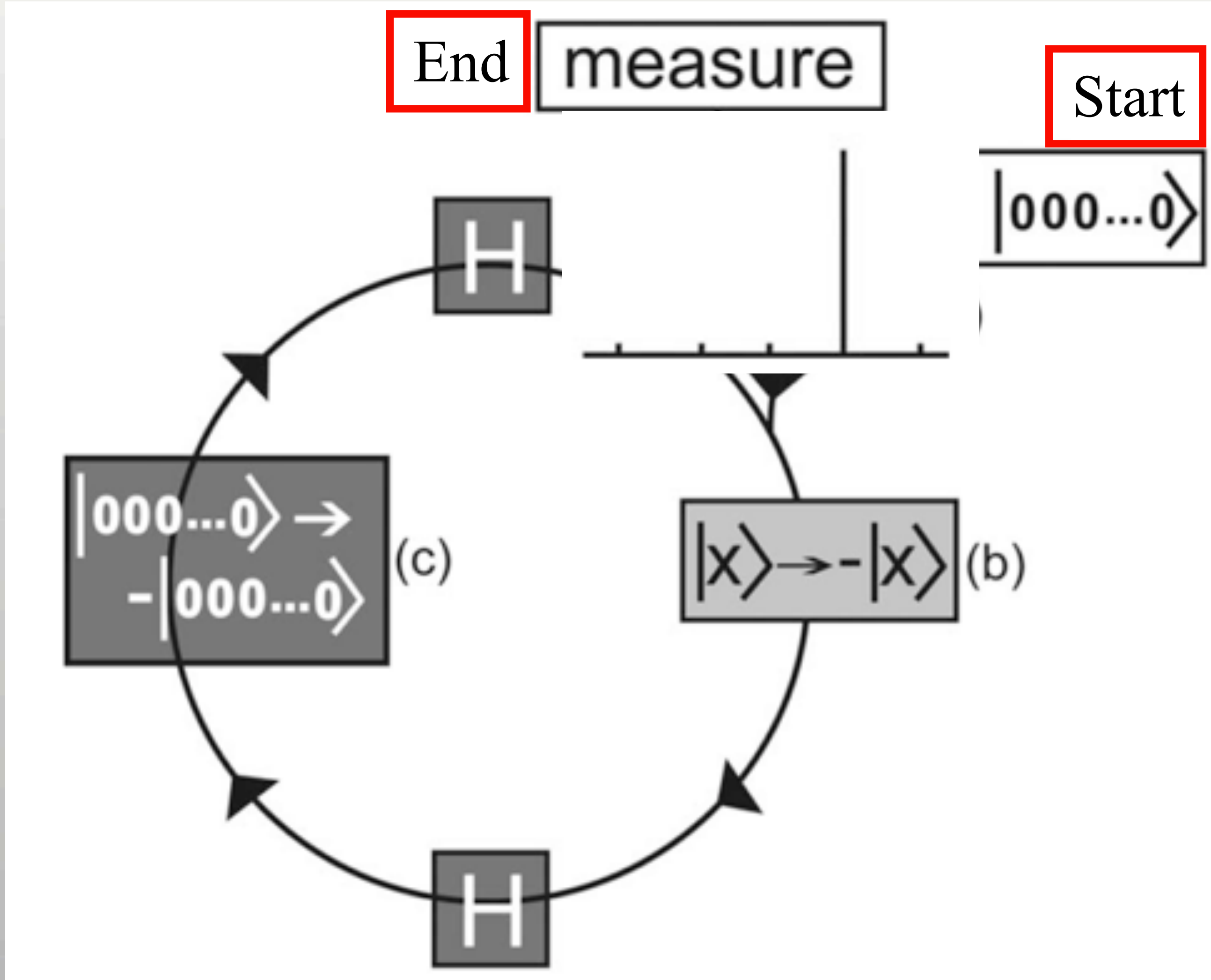
depicting Grover's algorithm

Brickmann et al: Phys. Rev. A. 72, 050306(R) (2005)



depicting Grover's algorithm

Brickmann et al: Phys. Rev. A. 72, 050306(R) (2005)

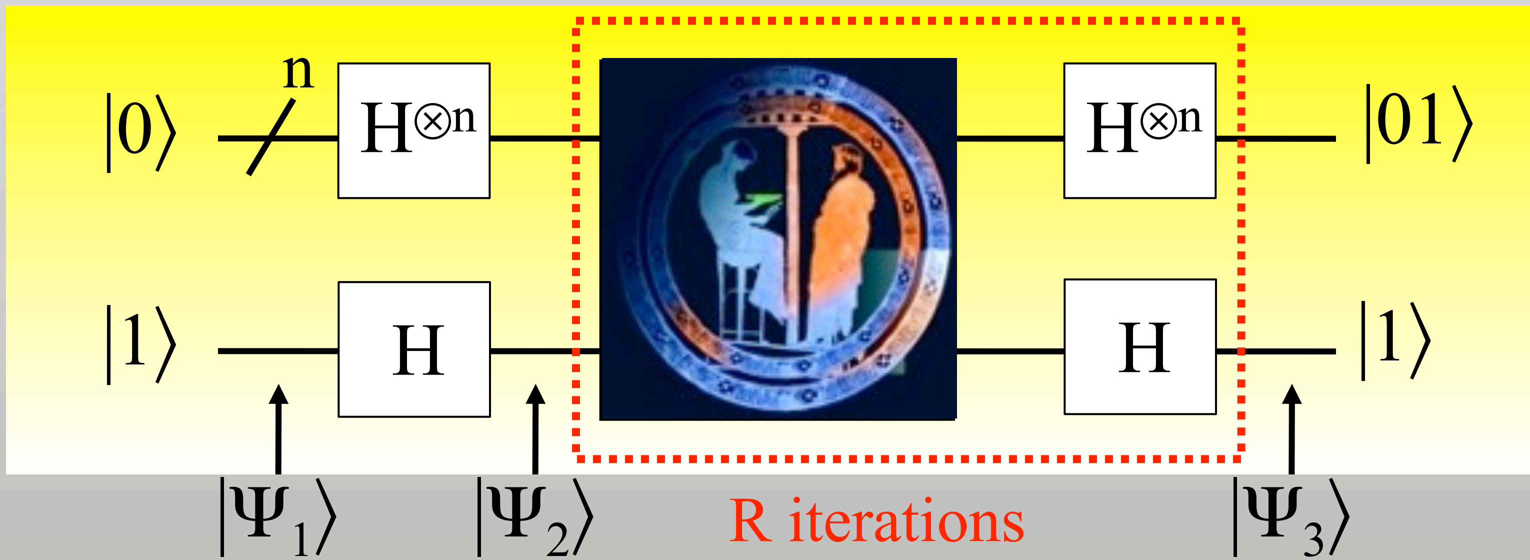


Quantum circuit

Data register: superposition of n=2 qubits **|x⟩** $|\Psi_1\rangle = |0\rangle$

$$H^{\otimes 2}|00\rangle = \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)$$

f(x) = 1



Target register: oracle qubit **|q⟩** prepared in $|\Psi_1\rangle = |1\rangle$

Oracle operator

$$U_O |x\rangle |q\rangle = |x\rangle |q \oplus f(x)\rangle$$

$|q\rangle$ is flipped, if $|x\rangle$ points to register with solution

$$|q_0\rangle := |\Psi_2\rangle = H|1\rangle = \frac{(|0\rangle - |1\rangle)}{\sqrt{2}}$$

$$|x\rangle \left[\frac{(|0\rangle - |1\rangle)}{\sqrt{2}}\right] \xrightarrow{U_O}$$

$$f(x) = 0: \; |x\rangle \left[\frac{(|0\rangle - |1\rangle)}{\sqrt{2}}\right]$$

$$f(x) = 1: \; |x\rangle \left[\frac{(|1\rangle - |0\rangle)}{\sqrt{2}}\right] = -|x\rangle \left[\frac{(|0\rangle - |1\rangle)}{\sqrt{2}}\right]$$

$$U_O |x\rangle |q_0\rangle = (-1)^{f(x)} |x\rangle |q_0\rangle$$

Grover's algorithm

Oracle qubit does not change: Look at data register only.

$$|\Psi_2\rangle = H^{\otimes 2}|00\rangle = \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} = \frac{1}{2}(|00\rangle + \boxed{|01\rangle} + |10\rangle + |11\rangle)$$

Grover's algorithm

Oracle qubit does not change: Look at data register only.

$$|\Psi_2\rangle = H^{\otimes 2}|00\rangle = \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} = \frac{1}{2}(|00\rangle + \boxed{|01\rangle} + |10\rangle + |11\rangle)$$

$$|\Psi_{3.1}\rangle = \mathbf{U_O}\,|\Psi_2\rangle = \frac{1}{2}(|00\rangle - |01\rangle + |10\rangle + |11\rangle)$$

Grover's algorithm

Oracle qubit does not change: Look at data register only.

$$|\Psi_2\rangle = H^{\otimes 2}|00\rangle = \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} = \frac{1}{2}(|00\rangle + \boxed{|01\rangle} + |10\rangle + |11\rangle)$$

$$|\Psi_{3.1}\rangle = \mathbf{U_O}|\Psi_2\rangle = \frac{1}{2}(|00\rangle - |01\rangle + |10\rangle + |11\rangle)$$

$$|\Psi_{3.2}\rangle = H^{\otimes 2}|\Psi_{3.1}\rangle = \frac{1}{4}(|00\rangle + |01\rangle + |10\rangle + |11\rangle - |00\rangle + |01\rangle - |10\rangle + |11\rangle + |00\rangle + |01\rangle - |10\rangle - |11\rangle + |00\rangle - |01\rangle - |10\rangle - |11\rangle)$$

$$= \frac{1}{2}(|00\rangle + |01\rangle - |10\rangle + |11\rangle)$$

Grover's algorithm

Oracle qubit does not change: Look at data register only.

$$|\Psi_2\rangle = H^{\otimes 2}|00\rangle = \frac{(|0\rangle + |1\rangle)}{\sqrt{2}}\frac{(|0\rangle + |1\rangle)}{\sqrt{2}} = \frac{1}{2}(|00\rangle + \boxed{|01\rangle} + |10\rangle + |11\rangle)$$

$$|\Psi_{3.1}\rangle = \mathbf{U_O}\,|\Psi_2\rangle = \frac{1}{2}(|00\rangle - |01\rangle + |10\rangle + |11\rangle)$$

$$|\Psi_{3.2}\rangle = H^{\otimes 2}|\Psi_{3.1}\rangle = \frac{1}{4}(|00\rangle + |01\rangle + |10\rangle + |11\rangle - |00\rangle + |01\rangle - |10\rangle + |11\rangle + |00\rangle + |01\rangle - |10\rangle - |11\rangle + |00\rangle - |01\rangle - |10\rangle - |11\rangle)$$

$$= \frac{1}{2}(|00\rangle + |01\rangle - |10\rangle + |11\rangle)$$

$$|\Psi_{3.3}\rangle = C_\pi|\Psi_{3.2}\rangle = (-1)^{\delta \mathbf{x0\text{-}1}}\,|\Psi_{3.2}\rangle = \frac{1}{2}(|00\rangle - |01\rangle + |10\rangle - |11\rangle)$$

Grover's algorithm

Oracle qubit does not change: Look at data register only.

$$|\Psi_2\rangle = H^{\otimes 2}|00\rangle = \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} = \frac{1}{2}(|00\rangle + \boxed{|01\rangle} + |10\rangle + |11\rangle)$$

$$|\Psi_{3.1}\rangle = U_O|\Psi_2\rangle = \frac{1}{2}(|00\rangle - |01\rangle + |10\rangle + |11\rangle)$$

$$|\Psi_{3.2}\rangle = H^{\otimes 2}|\Psi_{3.1}\rangle = \frac{1}{4}(|00\rangle + |01\rangle + |10\rangle + |11\rangle - |00\rangle + |01\rangle - |10\rangle + |11\rangle + |00\rangle + |01\rangle - |10\rangle - |11\rangle + |00\rangle - |01\rangle - |10\rangle - |11\rangle)$$

$$= \frac{1}{2}(|00\rangle + |01\rangle - |10\rangle + |11\rangle)$$

$$|\Psi_{3.3}\rangle = C_\pi|\Psi_{3.2}\rangle = (-1)^{\delta x0-1}|\Psi_{3.2}\rangle = \frac{1}{2}(|00\rangle - |01\rangle + |10\rangle - |11\rangle)$$

$$|\Psi_{3.4}\rangle = H^{\otimes 2}|\Psi_{3.3}\rangle =$$

Grover's algorithm

Oracle qubit does not change: Look at data register only.

$$|\Psi_2\rangle = H^{\otimes 2}|00\rangle = \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} = \frac{1}{2}(|00\rangle + \boxed{|01\rangle} + |10\rangle + |11\rangle)$$

$$|\Psi_{3.1}\rangle = U_O |\Psi_2\rangle = \frac{1}{2}(|00\rangle - |01\rangle + |10\rangle + |11\rangle)$$

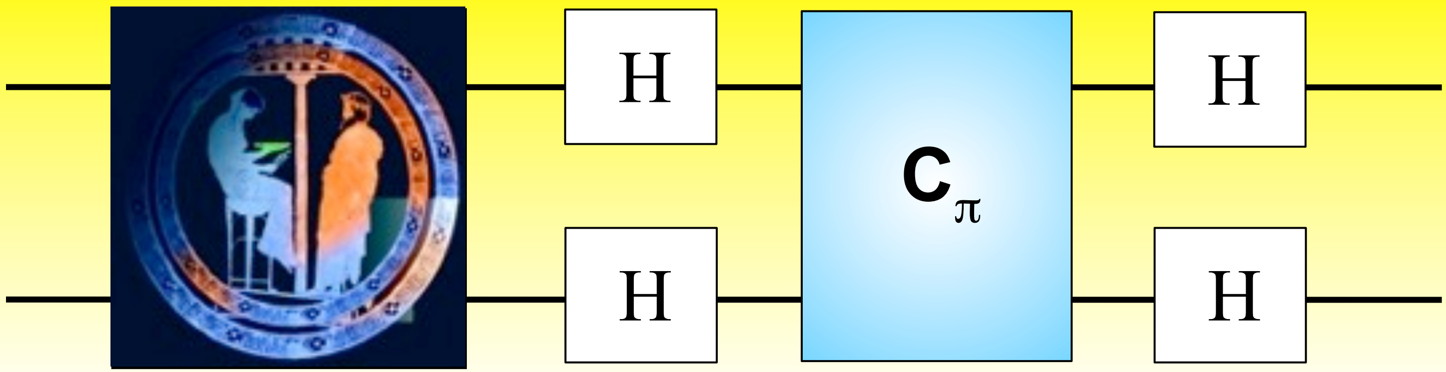
$$|\Psi_{3.2}\rangle = H^{\otimes 2}|\Psi_{3.1}\rangle = \frac{1}{4}(|00\rangle + |01\rangle + |10\rangle + |11\rangle - |00\rangle + |01\rangle - |10\rangle + |11\rangle + |00\rangle + |01\rangle - |10\rangle - |11\rangle + |00\rangle - |01\rangle - |10\rangle - |11\rangle)$$

$$= \frac{1}{2}(|00\rangle + |01\rangle - |10\rangle + |11\rangle)$$

$$|\Psi_{3.3}\rangle = C_\pi|\Psi_{3.2}\rangle = (-1)^{\delta x0-1}|\Psi_{3.2}\rangle = \frac{1}{2}(|00\rangle - |01\rangle + |10\rangle - |11\rangle)$$

$$|\Psi_{3.4}\rangle = H^{\otimes 2}|\Psi_{3.3}\rangle = |01\rangle$$

Grover's algorithm



$$U_O\,|x\rangle\,|q_0\rangle = (-1)^{f(x)}\,|x\rangle\,|q_0\rangle$$

$$H = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

$$C_\pi = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}$$

Geometrical analysis

$C_\pi = -\mathbb{1} + 2\,|0\rangle\langle 0|$ with $|\Psi_2\rangle = H^{\otimes n}|0\rangle$ and $\langle\Psi_2| = \langle 0|H^{\otimes n}$

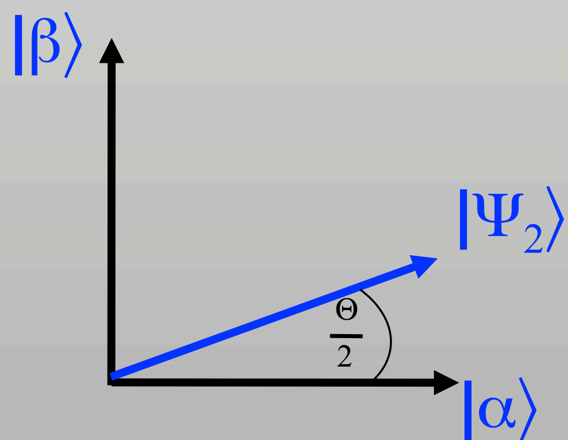
$$G = H^{\otimes n} C_\pi H^{\otimes n} U_O = H^{\otimes n}(2|0\rangle\langle 0| - \mathbb{1})H^{\otimes n} U_O = (2|\Psi_2\rangle\langle\Psi_2| - \mathbb{1})U_O$$

$$|\alpha\rangle = \frac{1}{\sqrt{N-M}} \sum_x (1-f(x))|x\rangle$$ Superpostion of “no-solutions”

$$|\beta\rangle = \frac{1}{\sqrt{M}} \sum_x f(x)|x\rangle$$ Superpostion of solutions

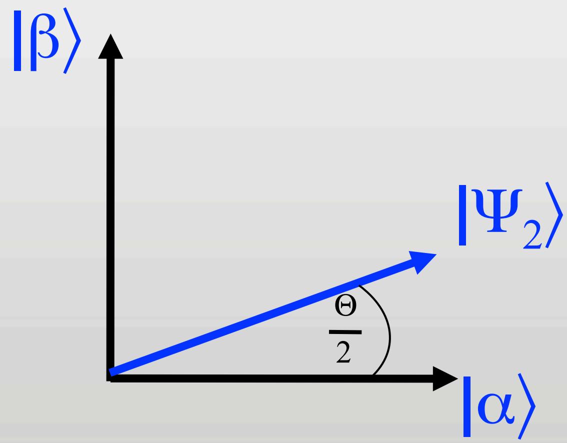
$$|\Psi_2\rangle = \sqrt{\frac{N-M}{N}}\,|\alpha\rangle + \sqrt{\frac{M}{N}}\,|\beta\rangle$$

$$= \cos\frac{\Theta}{2}|\alpha\rangle + \sin\frac{\Theta}{2}|\beta\rangle$$



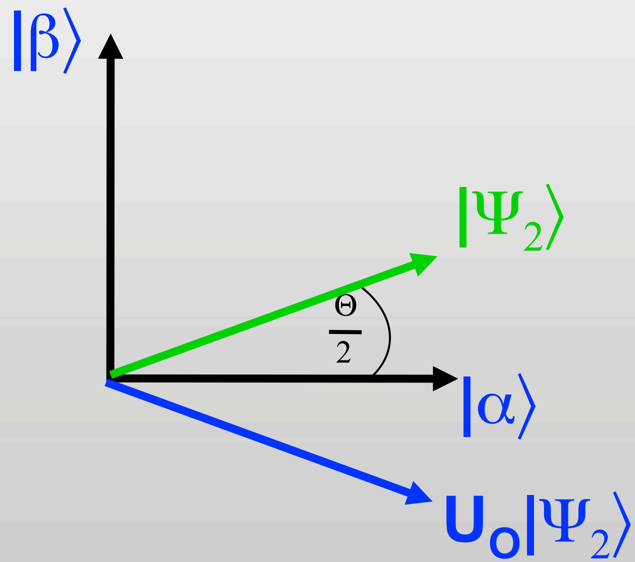
Geometrical analysis

$$U_O |\Psi_2\rangle = \cos\frac{\Theta}{2}|\alpha\rangle - \sin\frac{\Theta}{2}|\beta\rangle$$



Geometrical analysis

$$U_O |\Psi_2\rangle = \cos\frac{\Theta}{2}|\alpha\rangle - \sin\frac{\Theta}{2}|\beta\rangle$$

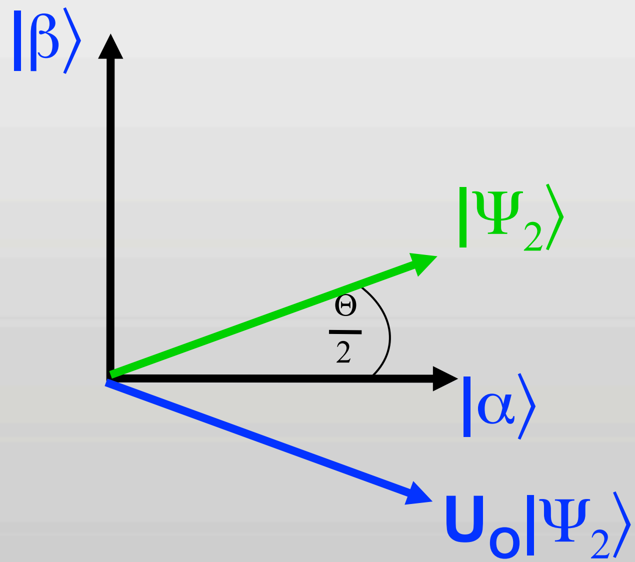


Geometrical analysis

$$U_O|\Psi_2\rangle = \cos\frac{\Theta}{2}|\alpha\rangle - \sin\frac{\Theta}{2}|\beta\rangle$$

$$(2|\Psi_2\rangle\langle\Psi_2| - \mathbb{1}) = |\Psi_2\rangle\langle\Psi_2| - (\mathbb{1} - |\Psi_2\rangle\langle\Psi_2|)$$

$$= P_2 - P_2^{\perp}$$



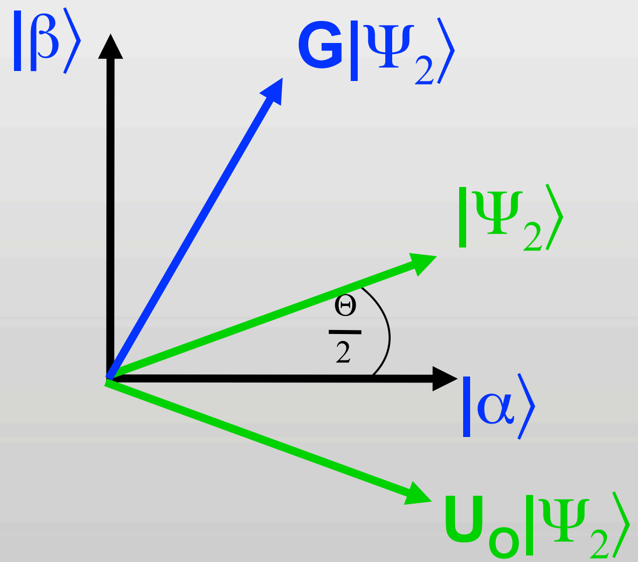
Geometrical analysis

$$U_O |\Psi_2\rangle = \cos\frac{\Theta}{2}|\alpha\rangle - \sin\frac{\Theta}{2}|\beta\rangle$$

$$(2|\Psi_2\rangle\langle\Psi_2| - \mathbb{1}) = |\Psi_2\rangle\langle\Psi_2| - (\mathbb{1} - |\Psi_2\rangle\langle\Psi_2|)$$

$$= P_2 - P_2^{\perp}$$

:reflexion at $|\Psi_2\rangle$



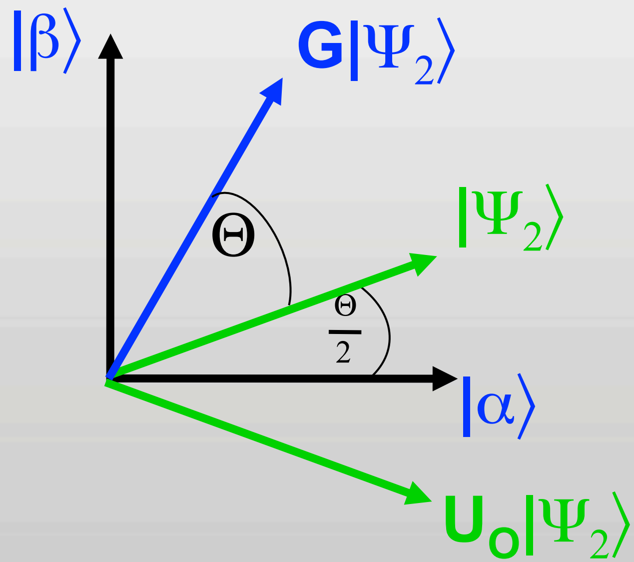
Geometrical analysis

$$U_O\,|\Psi_2\rangle = \cos\frac{\Theta}{2}|\alpha\rangle - \sin\frac{\Theta}{2}|\beta\rangle$$

$$(2|\Psi_2\rangle\langle\Psi_2| - \mathbb{1}) = |\Psi_2\rangle\langle\Psi_2| - (\mathbb{1} - |\Psi_2\rangle\langle\Psi_2|)$$

$$= P_2 - P_2^{\perp}$$

:reflexion at $|\Psi_2\rangle$



$$G\,|\Psi_2\rangle = \cos\frac{3\Theta}{2}|\alpha\rangle + \sin\frac{3\Theta}{2}|\beta\rangle$$

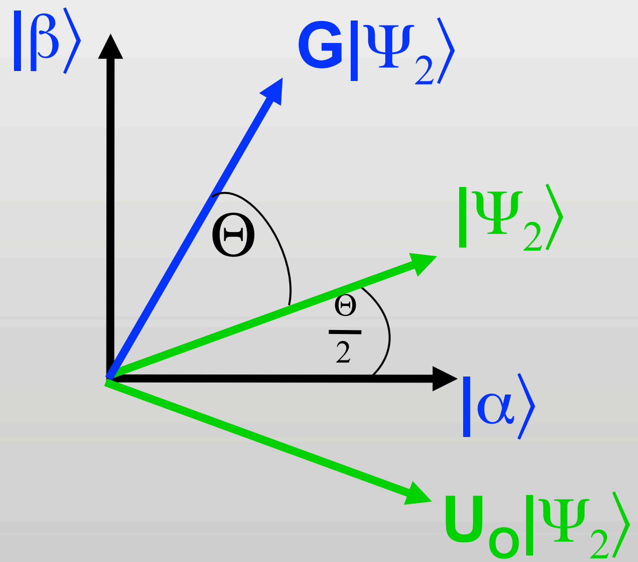
Geometrical analysis

$$U_O|\Psi_2\rangle = \cos\frac{\Theta}{2}|\alpha\rangle - \sin\frac{\Theta}{2}|\beta\rangle$$

$$(2|\Psi_2\rangle\langle\Psi_2| - \mathbb{1}) = |\Psi_2\rangle\langle\Psi_2| - (\mathbb{1} - |\Psi_2\rangle\langle\Psi_2|)$$

$$= P_2 - P_2^{\perp}$$

:reflexion at $|\Psi_2\rangle$



$$G|\Psi_2\rangle = \cos\frac{3\Theta}{2}|\alpha\rangle + \sin\frac{3\Theta}{2}|\beta\rangle$$

$$G = \begin{pmatrix} \cos\Theta & -\sin\Theta \\ \sin\Theta & \cos\Theta \end{pmatrix}$$

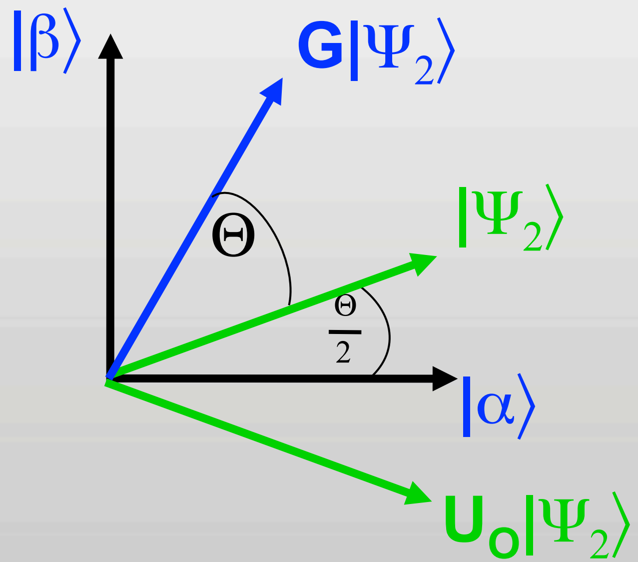
Geometrical analysis

$$U_O|\Psi_2\rangle = \cos\frac{\Theta}{2}|\alpha\rangle - \sin\frac{\Theta}{2}|\beta\rangle$$

$$(2|\Psi_2\rangle\langle\Psi_2| - \mathbb{1}) = |\Psi_2\rangle\langle\Psi_2| - (\mathbb{1} - |\Psi_2\rangle\langle\Psi_2|)$$

$$= P_2 - P_2^{\perp}$$

:reflexion at $|\Psi_2\rangle$



$$G|\Psi_2\rangle = \cos\frac{3\Theta}{2}|\alpha\rangle + \sin\frac{3\Theta}{2}|\beta\rangle$$

$$G^k|\Psi_2\rangle = \cos\frac{k+1}{2}\Theta\,|\alpha\rangle + \sin\frac{k+1}{2}\Theta|\beta\rangle$$

$$G = \begin{pmatrix} \cos\Theta & -\sin\Theta \\ \sin\Theta & \cos\Theta \end{pmatrix}$$

Iterations

Necessary number R of iterations is closest integer (CI) to

$$\frac{\frac{\pi}{2} - \frac{\Theta}{2}}{\Theta} = \frac{\pi - \Theta}{2\,\Theta} = \frac{\pi}{2\,\Theta} - \frac{1}{2}$$

$$R := CI\left(\frac{\pi}{2\,\Theta} - \frac{1}{2}\right) = \left(\frac{\pi}{4 \arcsin\sqrt{\frac{M}{N}}} - \frac{1}{2}\right) \leq \frac{\pi}{4}\sqrt{\frac{N}{M}}$$

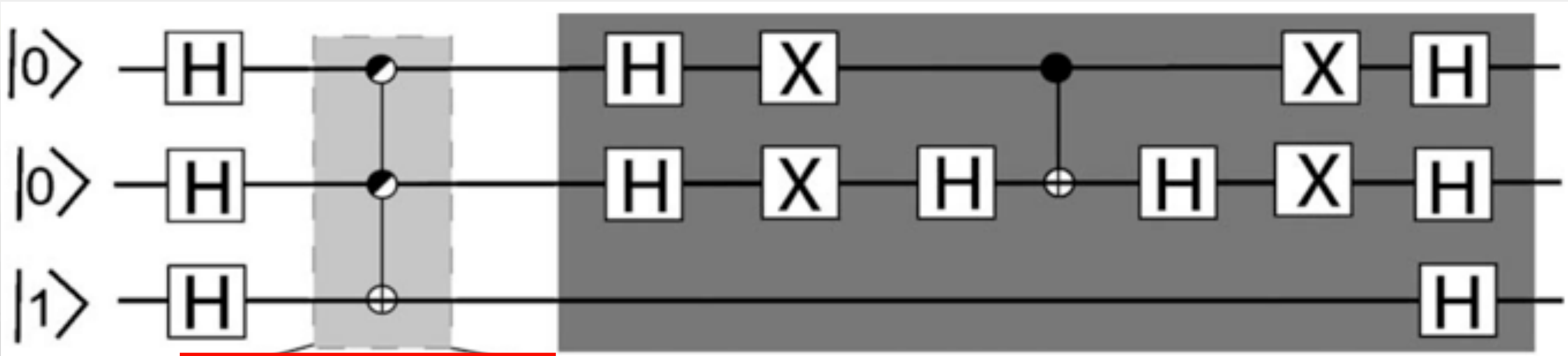
with $\frac{M}{N} << 1$ error probability $p \leq \sin^2\frac{\Theta}{2} = \frac{M}{N}$

- For more iterations than R, error increases

=> One needs to know number M of solutions.

Implementation quantum circuit

Brickmann et al: Phys. Rev. A. 72, 050306(R) (2005)



Toffoli gate: marks the state

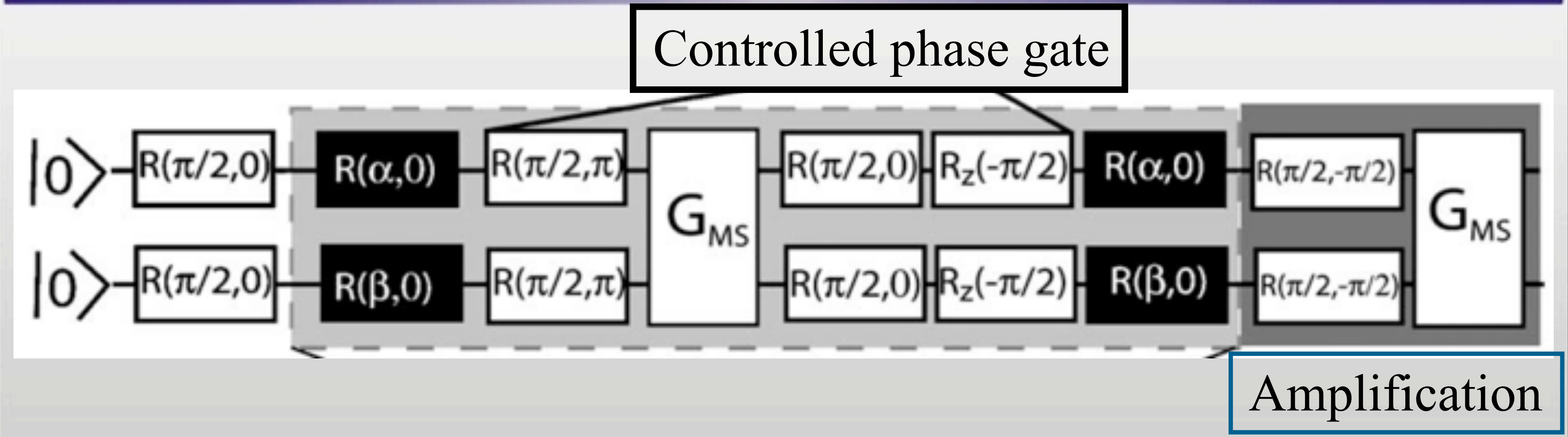
amplification of marked state

How to do this with ions?

1. use hyperfine states of $^{111}Cd^{+}$ ions
2. use microwaves to rotate spin states
3. Oracle qubit does not change: use vibration

Real implementation

Brickmann et al: Phys. Rev. A. 72, 050306(R) (2005)



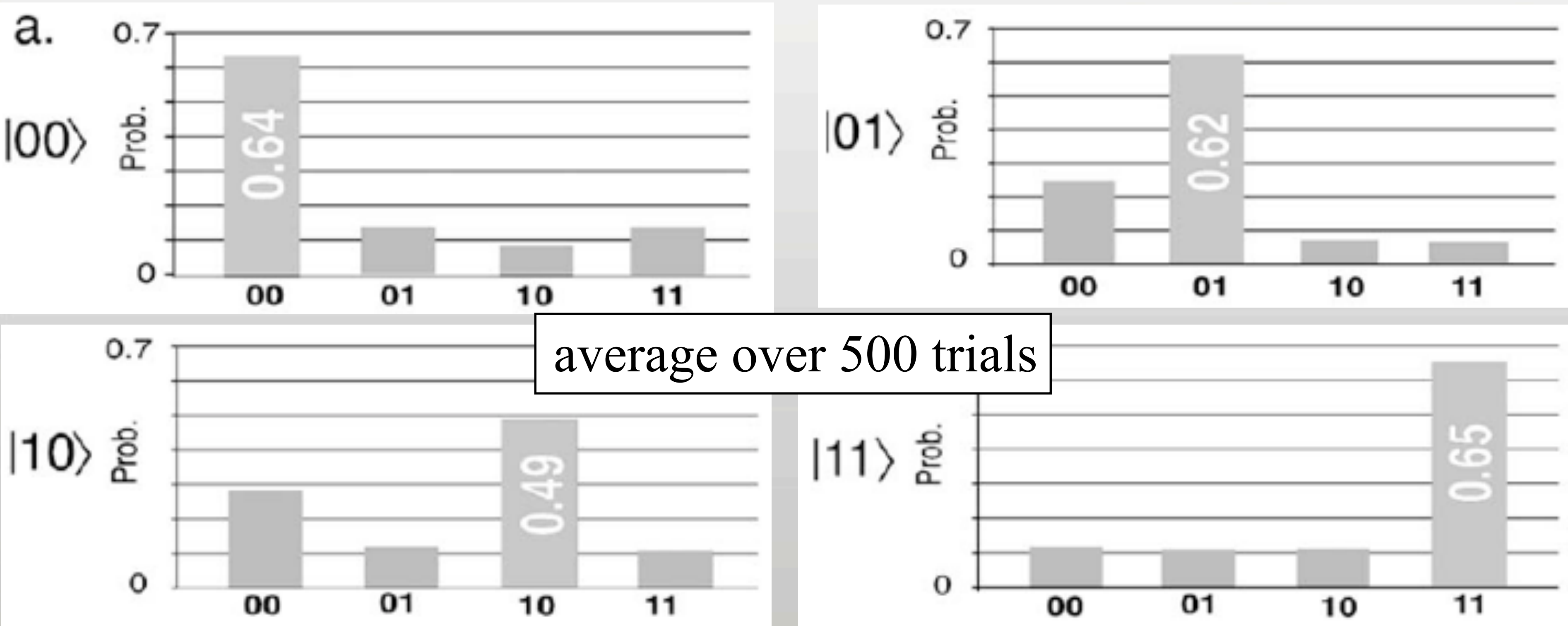
R : global microwave rotation combined with off-resonant laser pulses (induce Stark shift that leads to a phase shift π between the ions)

R_Z : phase rotation about the z axis

G_{MS} : Mølmer-Sørensen entangling gate
(developed for ions in traps -> phase shifts of lasers are compensated)

Grover algorithm: Fidelity

Brickmann et al: Phys. Rev. A. 72, 050306(R) (2005)



average over 500 trials

- time needed for the algorithm: ~ 380 μs (20 pulses)
- classical probability is 50% (1 query only)
- Grover should give 100%

ion trap quantum computing: a summary

- qubit representation
  - hyperfine states ($^{9}Be^{+}$, $^{43}Ca^{+}$)
  - electronic states ($^{40}Ca^{+}$)
  - vibrational modes
- qubit manipulation: laser irradiation
- initial state preparation:
  - Doppler and sideband cooling
- read-out: fluoresence

relaxation and operation

- electronic states:
  - energy relaxation time $T_1 \sim 1s$
  - phase relaxation time $T_2 \sim 10$ ms
  - gate operation time $T_{gate} \sim 200$ µs
  - $T_2/T_{gate} \sim 50$
- hyperfine states:
  - phase relaxation time $T_2 \sim 10s$
  - gate operation time $T_{gate} \sim 10$ µs
  - $T_2/T_{gate} \sim 10^6$

source: Homepage group A. Steane http://www.physics.ox.ac.uk/users/iontrap/news.html