Quantum Information with Solid-State Devices

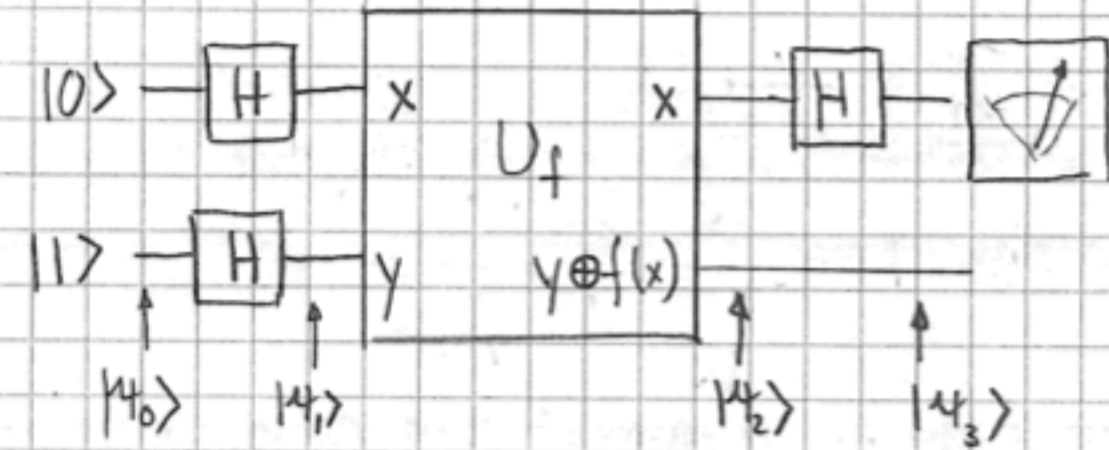
VO 141.246

Dr. Johannes Majer

Lecture 5

ATOMINSTITUT

TU WIEN



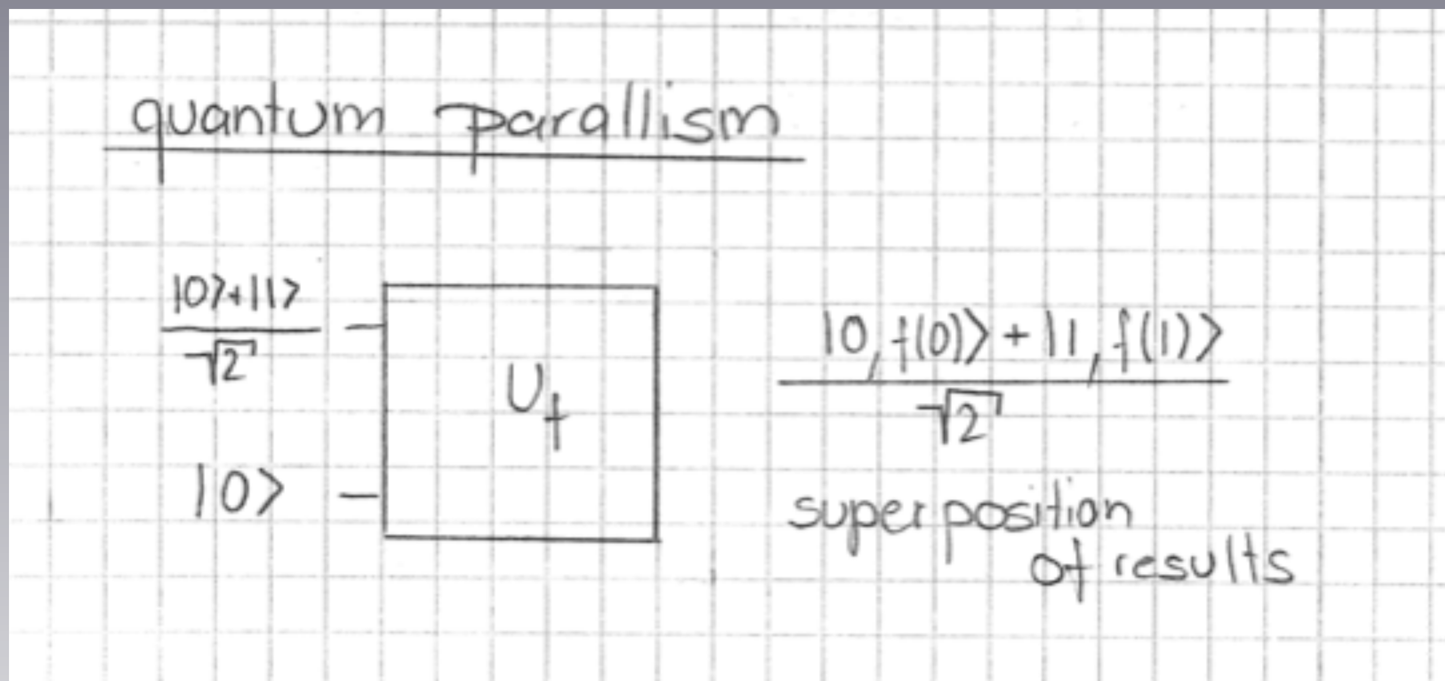
$$U_f : |x, y\rangle \longrightarrow |x, y \oplus f(x)\rangle$$

$\oplus$ is the addition modulo 2

$$U_{f(x)=0} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad U_{f(x)=1} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

$$U_{f(x)=x} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \quad U_{f(x)=1-x} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

CNOT


quantum parallism
|0>+|1> / √2
|0>
U_f
|0, f(0)> + |1, f(1)> / √2
superposition of results

Editor – /Users/johannesmajer2/Desktop/Matlab Lecture 4/DeutschAlgorithm.m

File Edit Text Go Cell Tools Debug Desktop Window Help

```matlab
% Demonstration of the Deutsch Algorithm

% 20-04-2010 © hannes@majer.ch


%% prepairing the elementary matrices

% pauli matrices
sigma_x = [0 1;...
           1 0];
sigma_y = [0 -i;...
           i  0];
sigma_z = [1  0;...
           0 -1];
id      = [1 0;...
           0 1];

% Hadamard
Had = [1  1;...
       1 -1]/sqrt(2);

% Controlled NOT
CNOT = [1 0 0 0;...
        0 1 0 0;...
        0 0 0 1;...
        0 0 1 0];

%% Implementation of Uf

% f(0)=0
% f(1)=0
Ufconst0 = [1 0 0 0;...
            0 1 0 0;...
            0 0 1 0;...
            0 0 0 1];

% f(0)=1
% f(1)=1
Ufconst1 = [0 1 0 0;...
            1 0 0 0;...
            0 0 0 1;...
            0 0 1 0];

% f(0)=0
% f(1)=1
```

TwoQubitGates.m | DeutschAlgorithm.m

script | Ln 15 | Col

Editor – /Users/johannesmajer2/Desktop/Matlab Lecture 4/DeutschAlgorithm.m

File Edit Text Go Cell Tools Debug Desktop Window Help

```
        1 -1]/sqrt(2);

% Controlled NOT
CNOT = [1 0 0 0;...
        0 1 0 0;...
        0 0 0 1;...
        0 0 1 0];

%% Implementation of Uf

% f(0)=0
% f(1)=0
Ufconst0 = [1 0 0 0;...
            0 1 0 0;...
            0 0 1 0;...
            0 0 0 1];

% f(0)=1
% f(1)=1
Ufconst1 = [0 1 0 0;...
            1 0 0 0;...
            0 0 0 1;...
            0 0 1 0];

% f(0)=0
% f(1)=1
Ufid     = [1 0 0 0;...
            0 1 0 0;...
            0 0 0 1;...
            0 0 1 0];

% f(0)=1
% f(1)=0
Ufneg    = [0 1 0 0;...
            1 0 0 0;...
            0 0 1 0;...
            0 0 0 1];

% choose one of the above gates
Uf = Ufneg;


%%

psi0 = kron([1 0]',[0 1]')
```

TwoQubitGates.m | DeutschAlgorithm.m

script | Ln 35 Col

Editor – /Users/johannesmajer2/Desktop/Matlab Lecture 4/DeutschAlgorithm.m

File Edit Text Go Cell Tools Debug Desktop Window Help

```
% f(0)=0
% f(1)=0
Ufconst0 = [1 0 0 0;...
            0 1 0 0;...
            0 0 1 0;...
            0 0 0 1];

% f(0)=1
% f(1)=1
Ufconst1 = [0 1 0 0;...
            1 0 0 0;...
            0 0 0 1;...
            0 0 1 0];

% f(0)=0
% f(1)=1
Ufid     = [1 0 0 0;...
            0 1 0 0;...
            0 0 0 1;...
            0 0 1 0];

% f(0)=1
% f(1)=0
Ufneg    = [0 1 0 0;...
            1 0 0 0;...
            0 0 1 0;...
            0 0 0 1];

% choose one of the above gates
Uf = Ufneg;


%%

psi0 = kron([1 0]',[0 1]')

psi1 = kron(Had,id)*kron(id,Had)*psi0

psi2 = Uf*psi1

psi3 = kron(Had,id)*psi2

sz = psi3'*kron(sigma_z,id)*psi3

(1-sz)/2
```

TwoQubitGates.m   DeutschAlgorithm.m

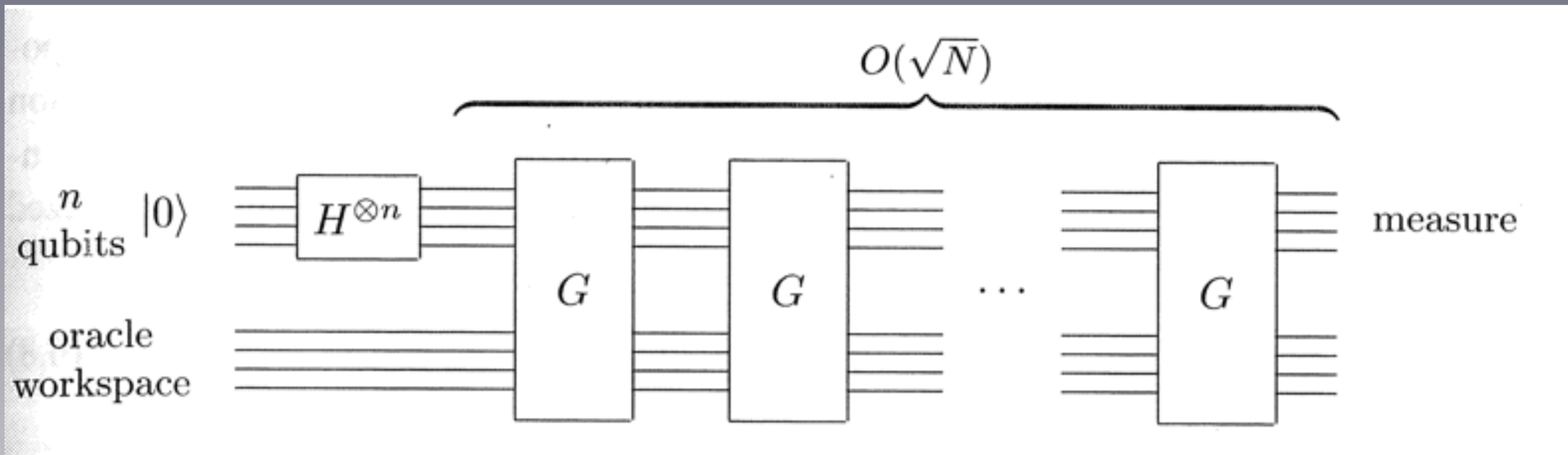
script   Ln 64   Col

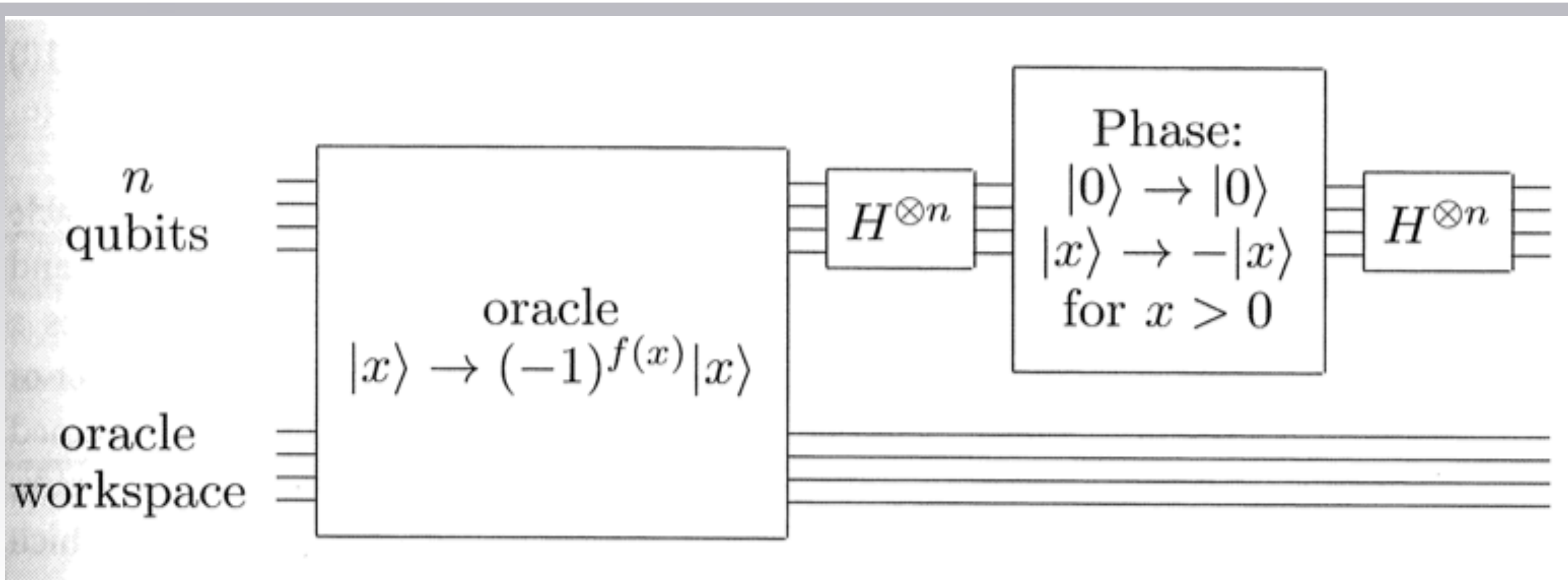
```
%% Implementation of Uf

%constant function

Ufconst0 = [1 0 0 0 0 0 0 0;...
            0 1 0 0 0 0 0 0;...
            0 0 1 0 0 0 0 0;...
            0 0 0 1 0 0 0 0;...
            0 0 0 0 1 0 0 0;...
            0 0 0 0 0 1 0 0;...
            0 0 0 0 0 0 1 0;...
            0 0 0 0 0 0 0 1];

Ufconst1 = [0 1 0 0 0 0 0 0;...
            1 0 0 0 0 0 0 0;...
            0 0 0 1 0 0 0 0;...
            0 0 1 0 0 0 0 0;...
            0 0 0 0 0 1 0 0;...
            0 0 0 0 1 0 0 0;...
            0 0 0 0 0 0 0 1;...
            0 0 0 0 0 0 1 0];


% balanced functions

% f(0,0)=1
% f(0,1)=0
% f(1,0)=0
% f(1,1)=1
Ufb1001  = [0 1 0 0 0 0 0 0;...
            1 0 0 0 0 0 0 0;...
            0 0 1 0 0 0 0 0;...
            0 0 0 1 0 0 0 0;...
            0 0 0 0 1 0 0 0;...
            0 0 0 0 0 1 0 0;...
            0 0 0 0 0 0 0 1;...
            0 0 0 0 0 0 1 0];

% f(0,0)=1
% f(0,1)=1
% f(1,0)=0
% f(1,1)=0
Ufb1100  = [0 1 0 0 0 0 0 0;...
            1 0 0 0 0 0 0 0;...
            0 0 0 1 0 0 0 0;...
            0 0 1 0 0 0 0 0;
```

Deutsch-Josza

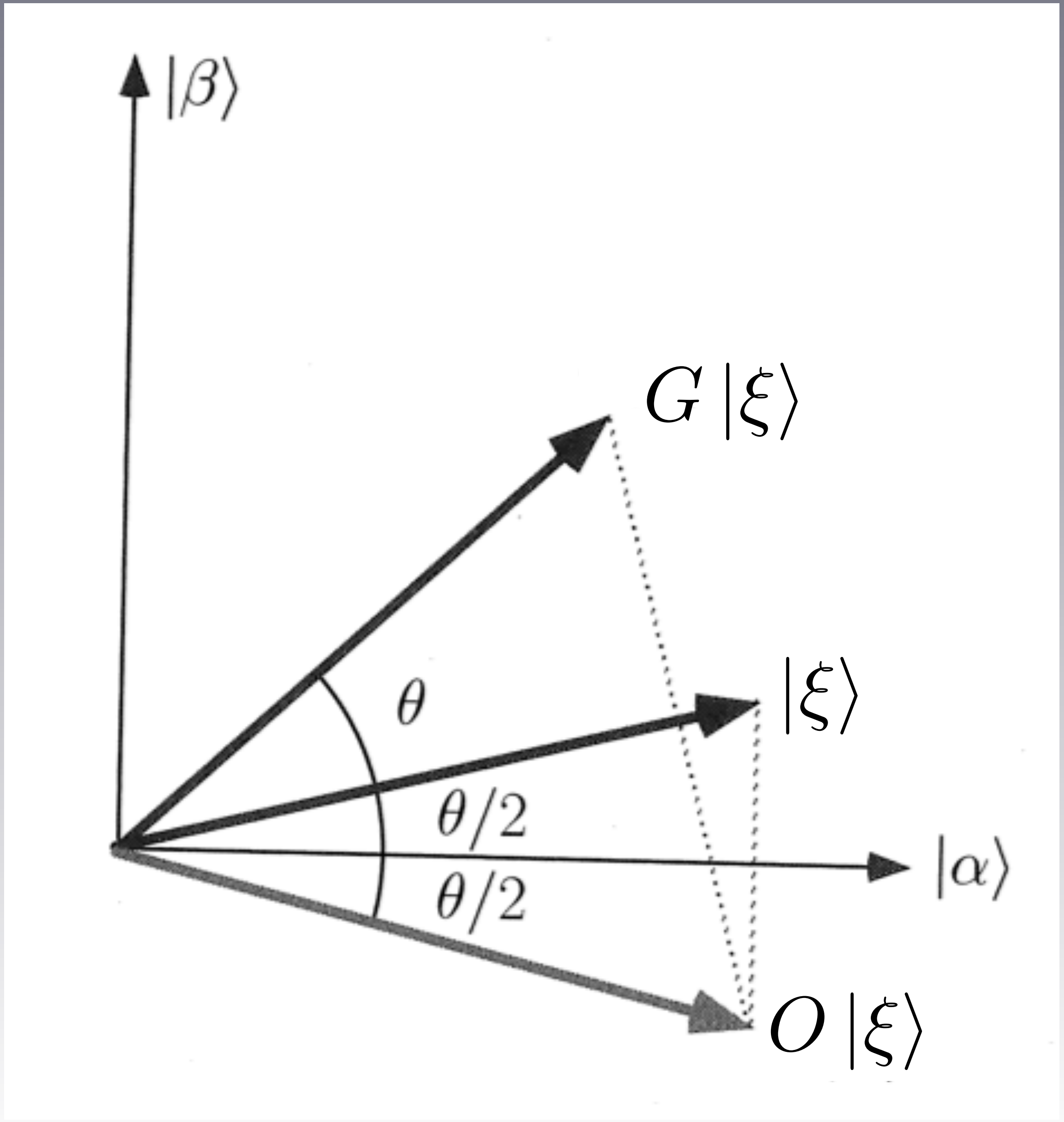
```
%%

psi0 = kron([1 0 0 0]',[0 1]')

psi1 = kron(Had2,id)*kron(id2,Had)*psi0

psi2 = Uf*psi1

psi3 = kron(Had2,id)*psi2

sz1 = psi3'*kron(kron(sigma_z,id),id)*psi3
sz2 = psi3'*kron(kron(id,sigma_z),id)*psi3
```

Grover algorithm

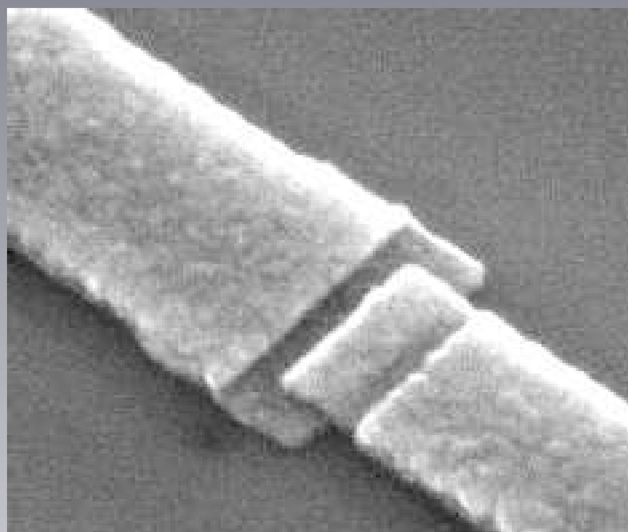




Grover algorithm



II Superconducting Electronics



Josephson junction
superconductors
tunnel junctions
Josephson equations
SQUID