How to Build a Quantum Computer doi.org/cr58

Barry C. Sanders

Table of Contents

1. Introduction
2. Entanglement generation
3. Implementations
4. Semiconductor
5. Superconductor

@bcsanders

barry-sanders-69b5519

wxid_8fjd7iane0eu12

http:// iqst.ca/people/peoplepage.php?id=4

ORCID 0000-0002-8326-8912

Basics of Computing

Universal Computer

Solves output from input data $\boldsymbol{b} \in \{0,1\}^*$ by executing a program implemented according to a machine-specific instruction set.

Types of computational problems

Decision $f : \{0,1\}^* \to \{0,1\}$, function, promise, search, optimization, ...

Decompose $f : \{0,1\}^* \to \{0,1\}$ into Boolean logic operations

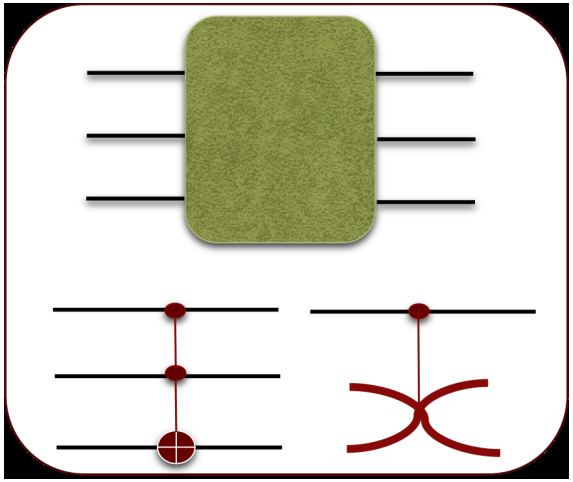
Execute f via concatenating unary $f : \{0,1\} \to \{0,1\}$ & binary $f : \{0,1\}^2 \to \{0,1\}$ operations such as $\wedge$, $\vee$ & $\neg$.

x	y	$x \wedge y$	$x \vee y$
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	1

Reversible computing

Embed $f : \{0,1\}^* \to \{0,1\}$ into reversible logic and decomposing

Reversible $F : \{0,1\}^* \times \{0,1\} \to \{0,1\}^* \times \{0,1\} : (x, y) \mapsto (x, y \oplus f(x))$ decomposable into CCNOT or CSWAP gates.



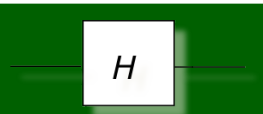
Q Computing

Same problems; different instructions

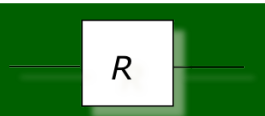
Solves $f : \{0,1\}^* \to \{0,1\}$ but maps to Q information to process according to Q logic and measures to yield output.

Qbits $|\Psi\rangle = \sum_{\boldsymbol{b}\in\{0,1\}^n} \Psi_{\boldsymbol{b}} |\boldsymbol{b}\rangle \in \mathbb{C}^{2^n}$ & Gates $G : \mathbb{C}^{2^k} \to \mathbb{C}^{2^k}$

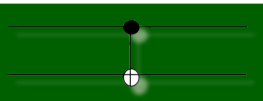
For n Qbits plus k-Qbit gates with k fixed size independent of n.

Hadamard gate: 

$$H : |b\rangle \mapsto |\bar{b}\rangle := \frac{|0\rangle + (-1)^b |1\rangle}{\sqrt{2}}$$

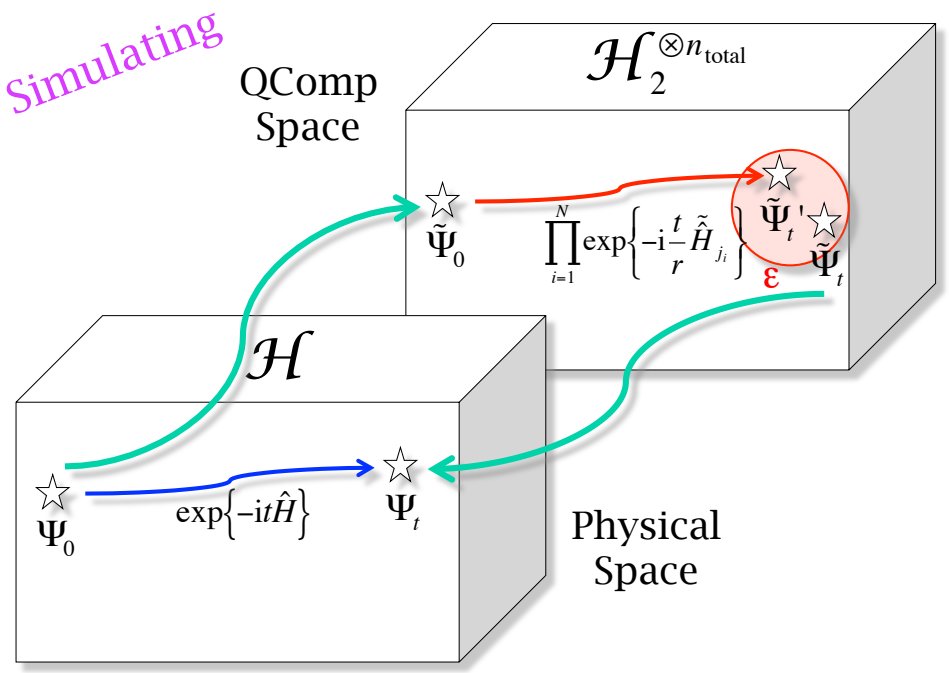
T ('magic') gate: 

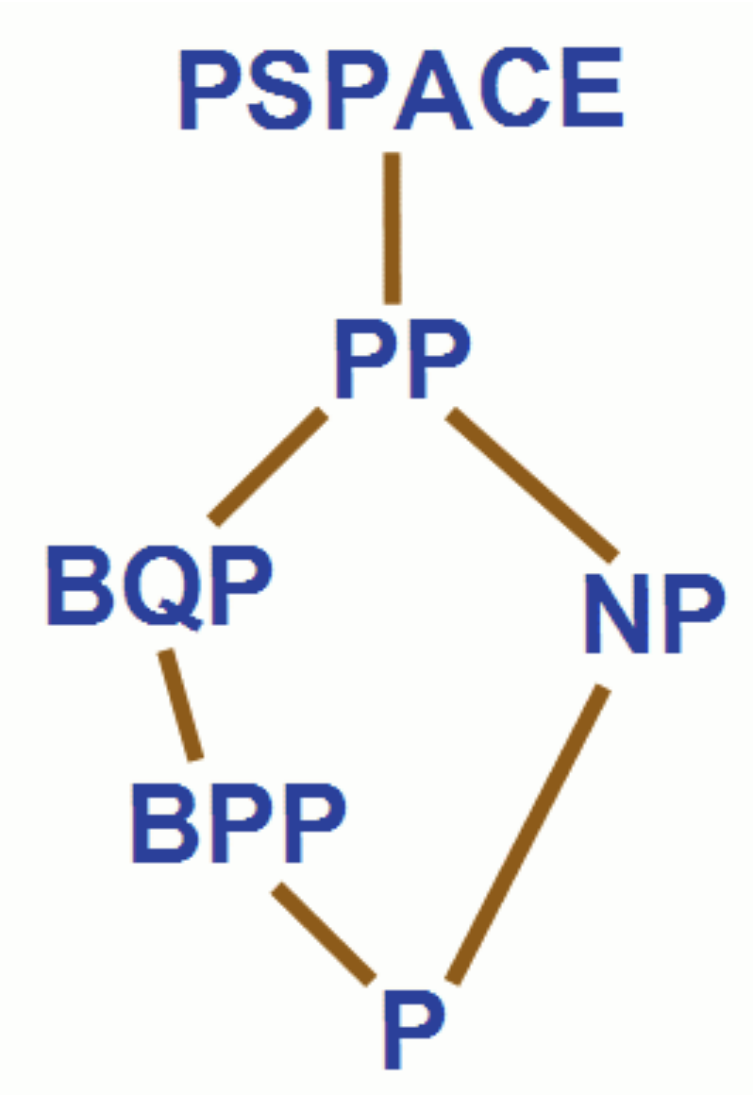
$$T : |b\rangle \mapsto e^{ib\pi/4} |b\rangle.$$

CNOT gate: 

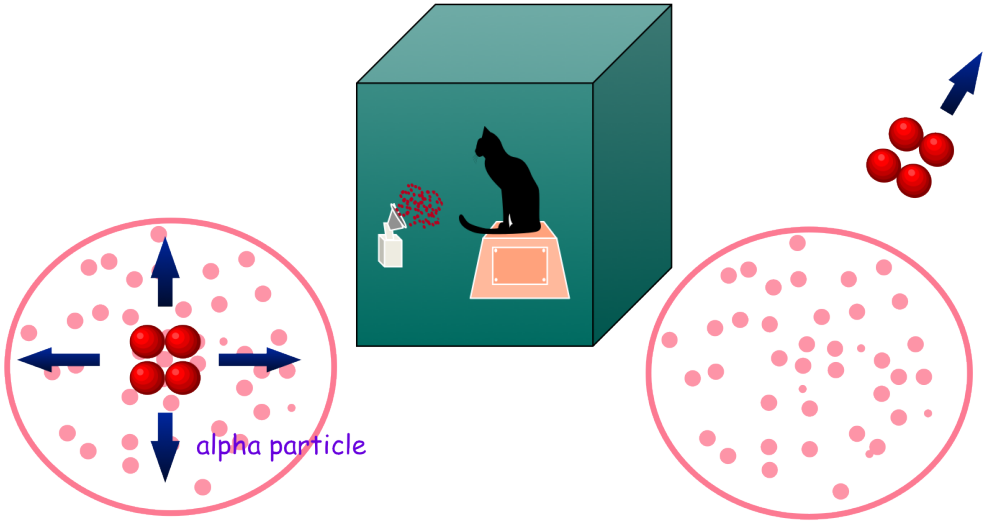
$$\text{CNOT} : |bb'\rangle \mapsto |b\ b' \oplus b\rangle$$

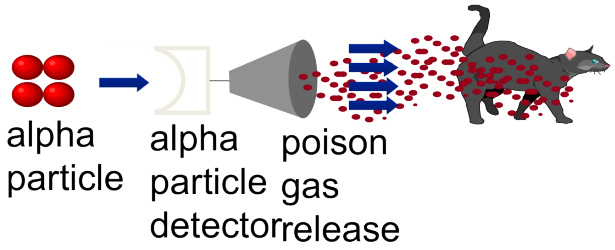
Q information processing & complexity

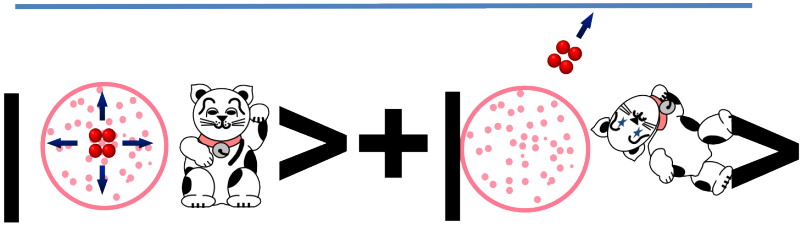


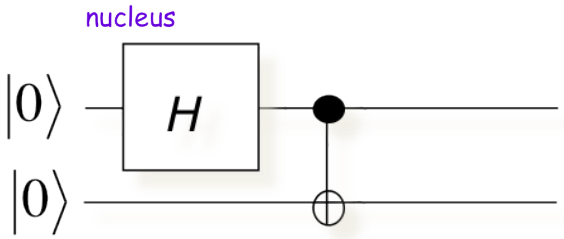


Schrödinger's cat schematic



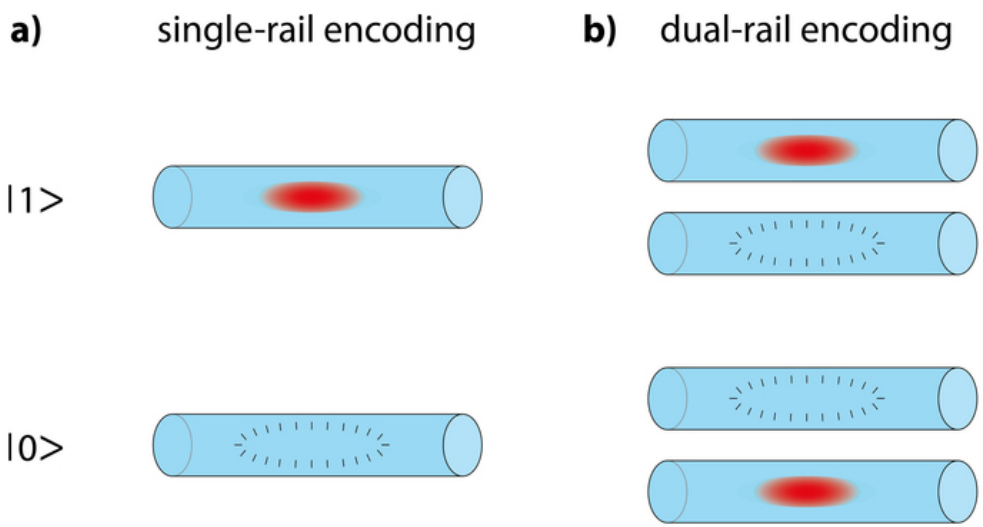


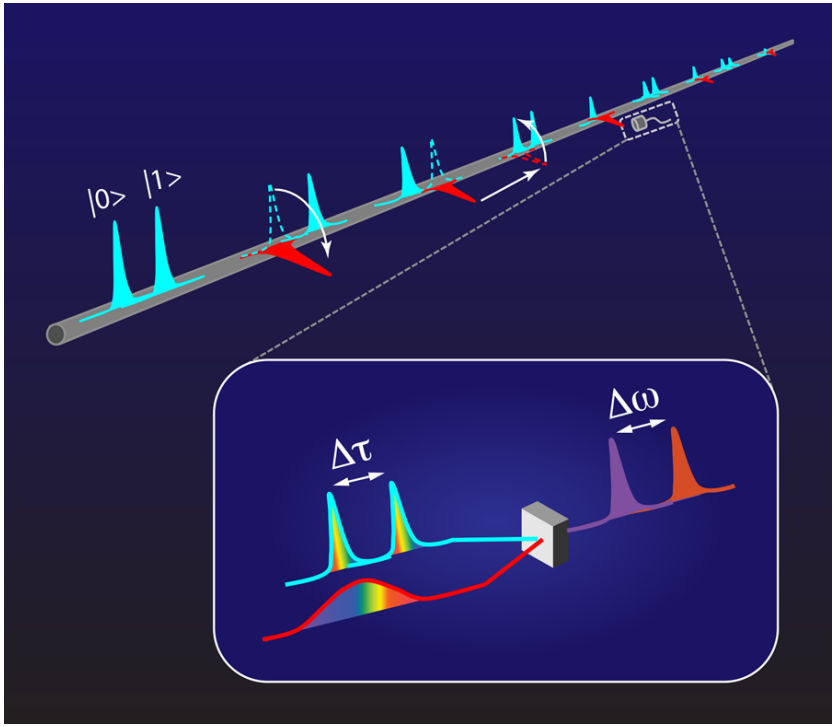


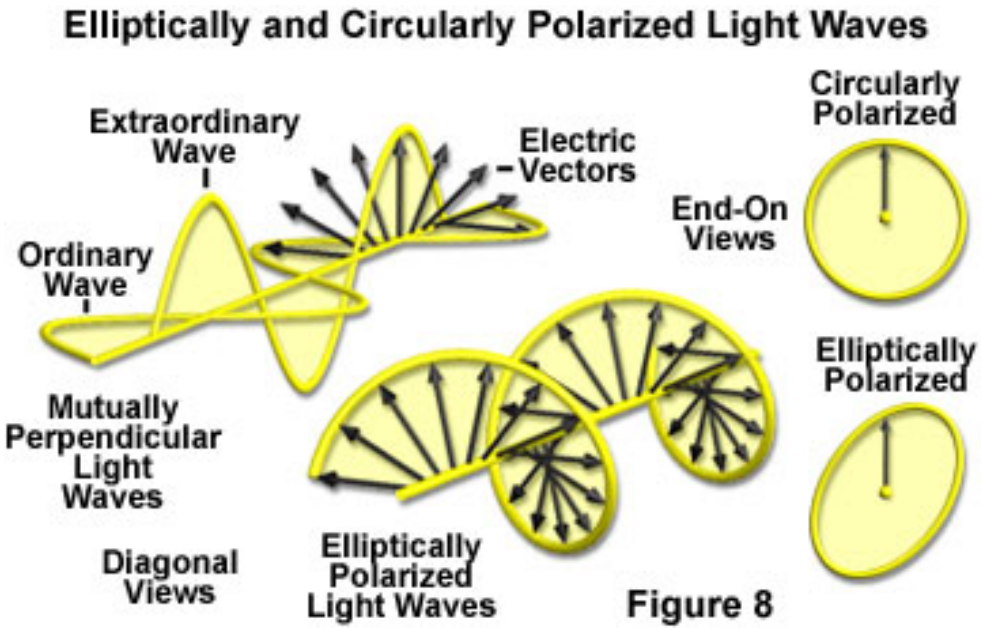


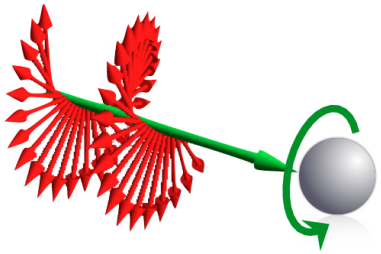
$$\frac{|00\rangle + |11\rangle}{\sqrt{2}}$$

Encoding Q Information into Light

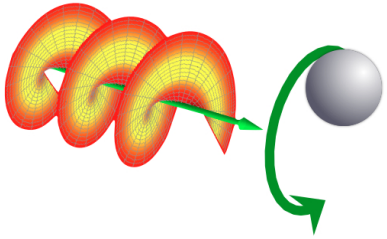








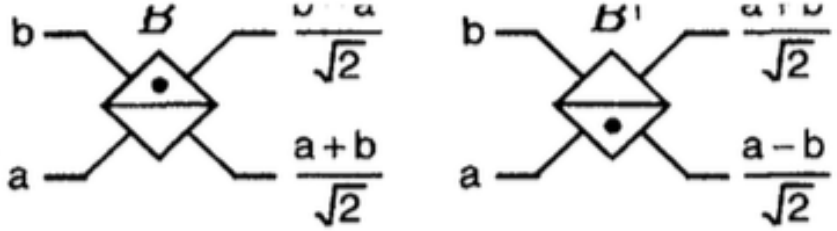
SAM interaction



OAM interaction

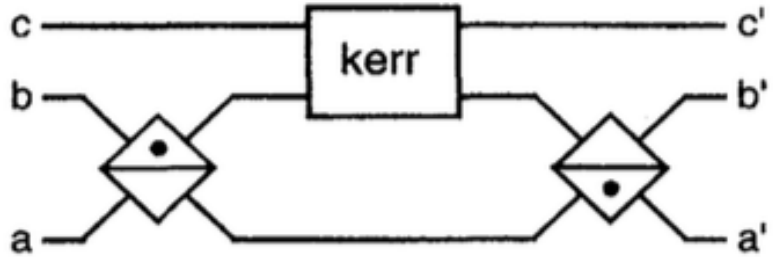
Dual-Rail Nonlinear Q Optical Computer doi.org/fkkvp6

quantum beamsplitter



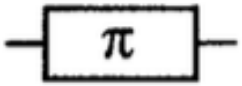
$$B = \exp\left[\frac{\pi}{4}\left(a^\dagger b - a b^\dagger\right)\right]$$

quantum Fredkin gate



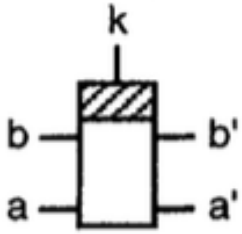
$$F = \exp\left[\frac{\chi}{2} c^\dagger c \left(a^\dagger b - a b^\dagger\right)\right]$$

phase modulator



$$S = \exp\left[i\pi a^\dagger a\right]$$

classical switch



k=0 : a'=a, b'=b
k=1 : a'=b, b'=a

Knill-Laflamme-Milburn doi.org/cxdhtn

A scheme for efficient quantum computation with linear optics

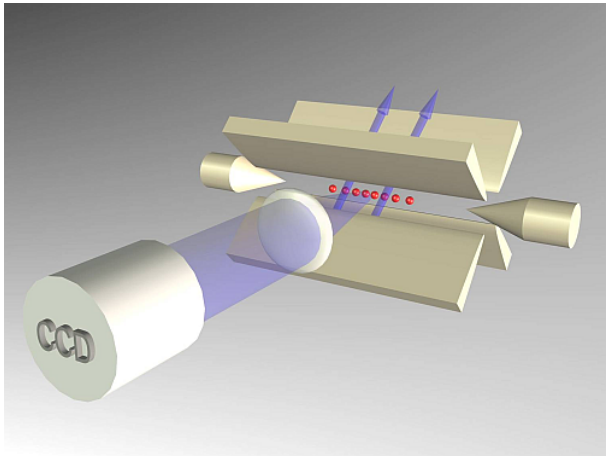
E. Knill*, **R. Laflamme*** & **G. J. Milburn**†

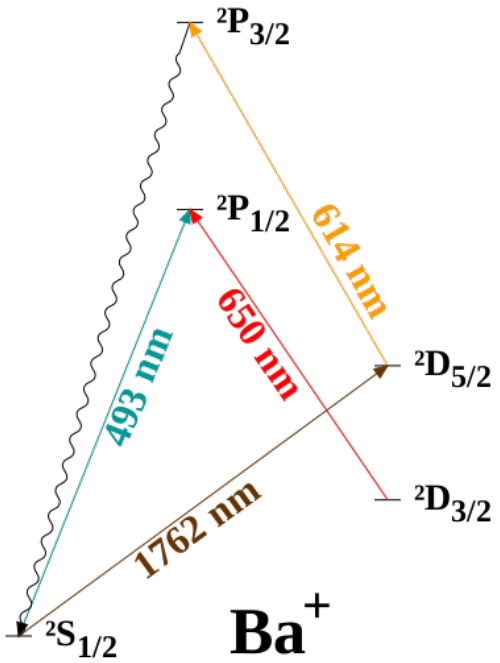
* *Los Alamos National Laboratory, MS B265, Los Alamos, New Mexico 87545, USA*
† *Centre for Quantum Computer Technology, University of Queensland, St. Lucia, Australia*

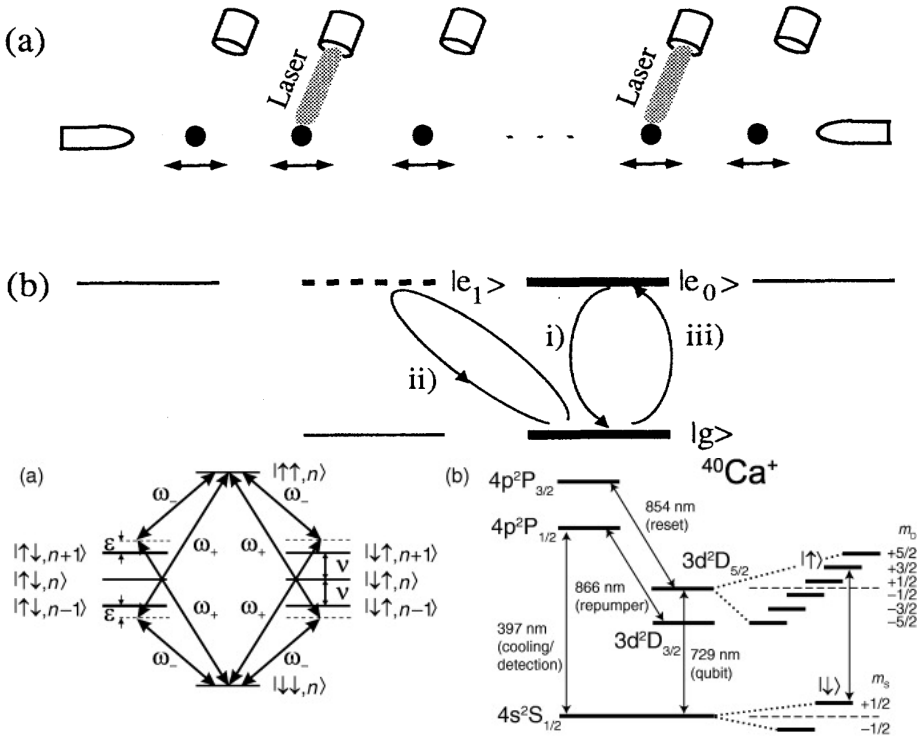
Quantum computers promise to increase greatly the efficiency of solving problems such as factoring large integers, combinatorial optimization and quantum physics simulation. One of the greatest challenges now is to implement the basic quantum-computational elements in a physical system and to demonstrate that they can be reliably and scalably controlled. One of the earliest proposals for quantum computation is based on implementing a quantum bit with two optical modes containing one photon. The proposal is appealing because of the ease with which photon interference can be observed. Until now, it suffered from the requirement for non-linear couplings between optical modes containing few photons. Here we show that efficient quantum computation is possible using only beam splitters, phase shifters, single photon sources and photo-detectors. Our methods exploit feedback from photo-detectors and are robust against errors from photon loss and detector inefficiency. The basic elements are accessible to experimental investigation with current technology.

© 2001 Macmillan Magazines Ltd
NATURE | VOL 409 | 4 JANUARY 2001 | www.nature.com

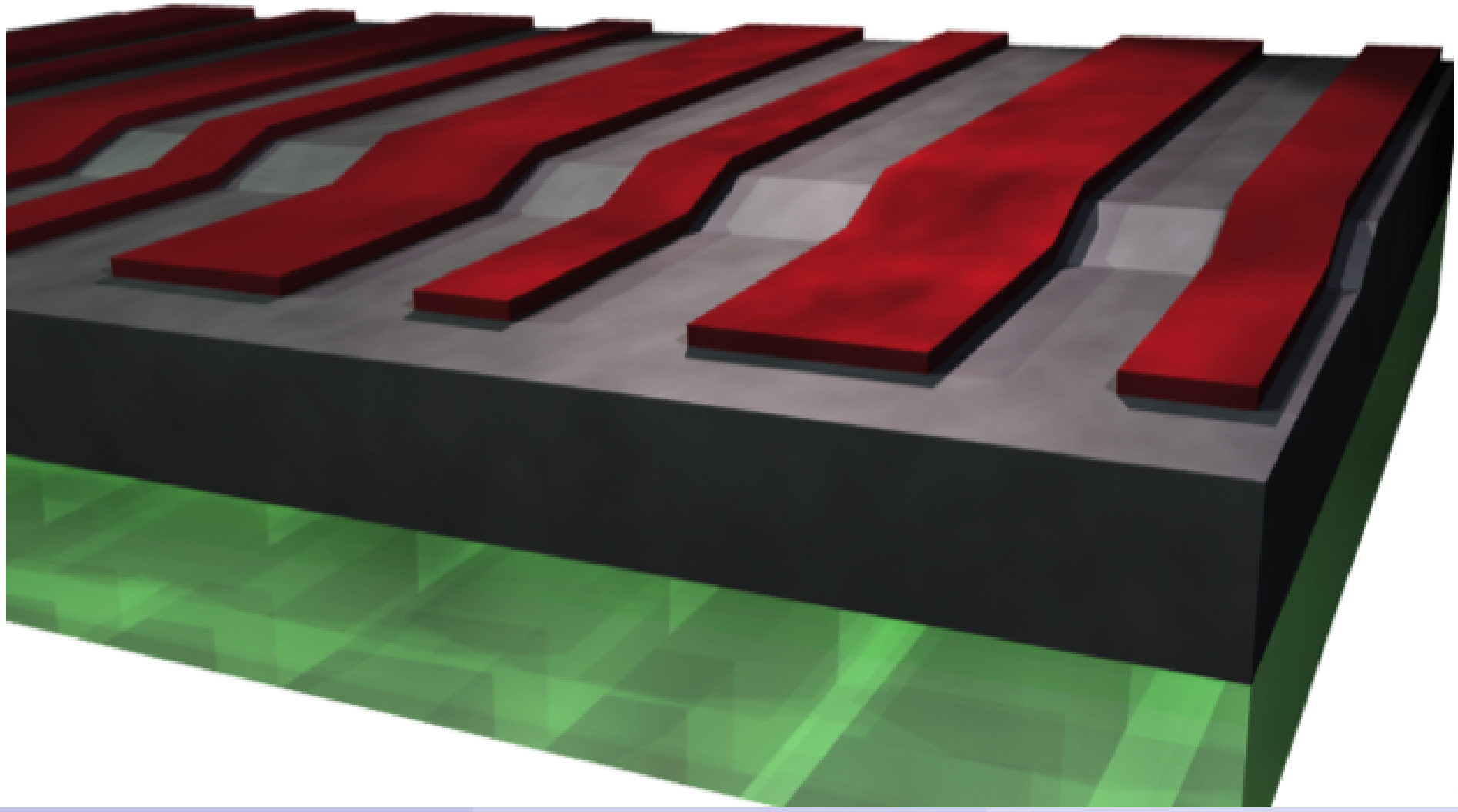
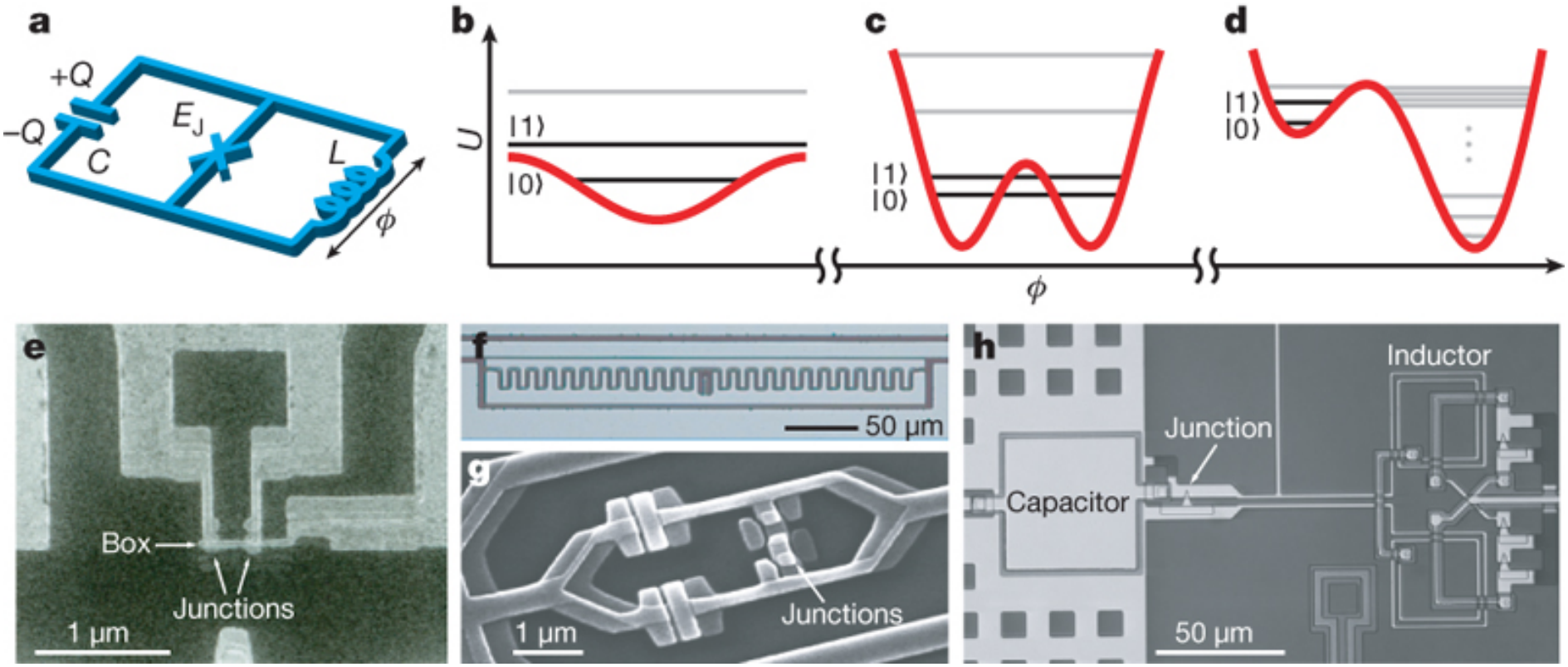
Linear Ion Trap







Q Computer Technology: P in Si doi.org/cf32zn



Superconducting Qbit