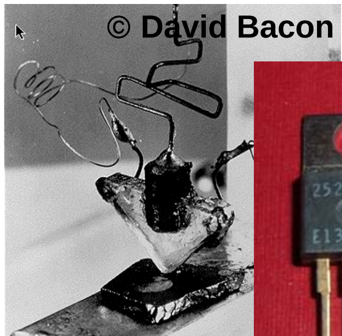
# Quantum Computing for Computer Architects

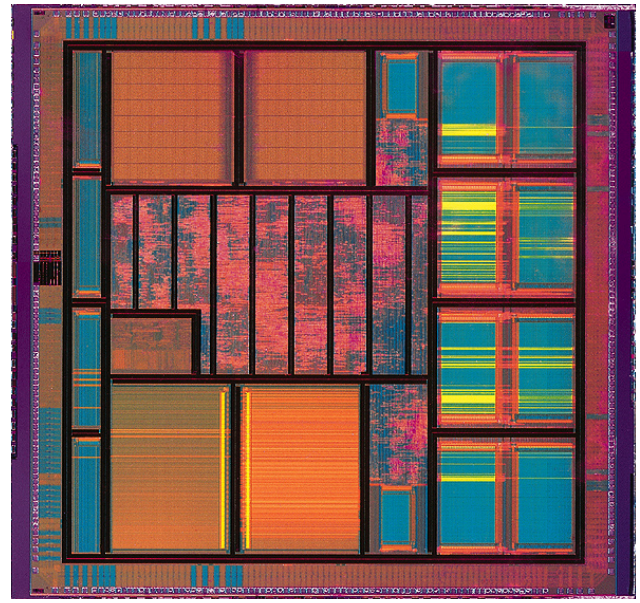
Tzvetan S. Metodi, Darshan D. Thaker, and Frederic T. Chong

# Modern Computers

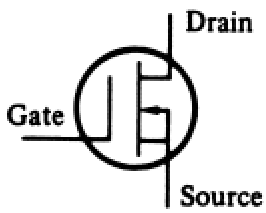
transistors




1954

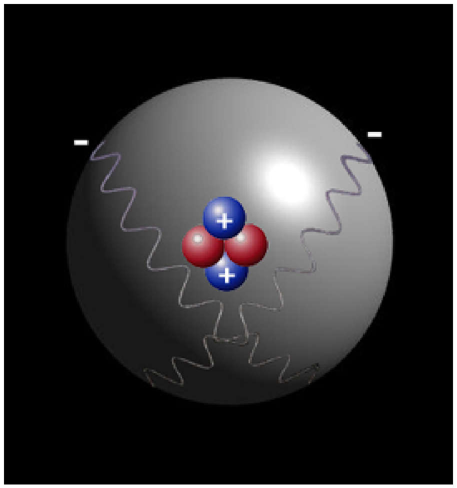


integrated circuits



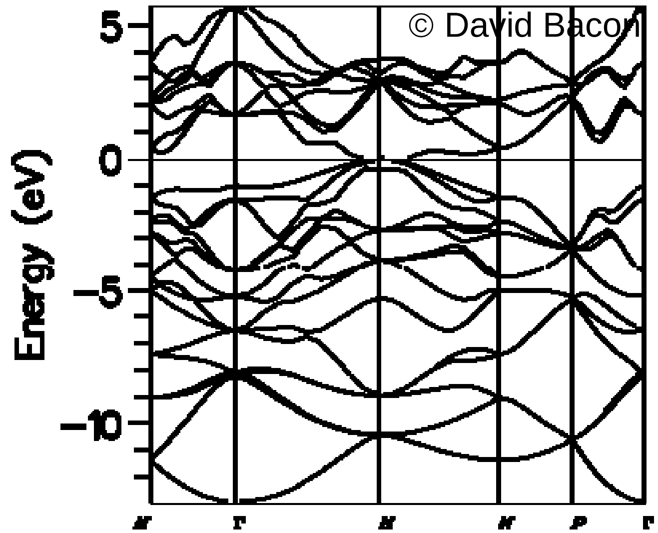

reliable and deterministic gates (although this is changing)

# Quantum Computer?



Single Atom

Quantum Data Bit (QUBIT)




qubit states

$\boxed{0}$ = ground state

$\boxed{1}$ = excited state

**In reality the state of a qubit is in a probabilistic superposition between the two possible states.**

# Why Quantum Computation?

- **Factoring**
  - Factoring: given N=pq, find p and q.
  - Best algorithm $2^{O(n^{(1/3)})}$, n -number of digits.
  - **The RSA cryptosystem is based on hardness of factoring.**
  - **Polynomial time quantum algorithm [Shor, 1994]**
  - Similar quantum algorithm solves discrete log problem.

**Classical Factoring Experiment:** Cavallar in 2000 has demonstrated the factorization of a 512-bit number in seven calendar months on 300 fast workstations, two SGI Origin 2000 computers, and one Cray C916 Supercomputer.

# Why Quantum Computation?

- Factoring
  - Factoring: given N=pq, find p and q.
  - Best algorithm $2^{O(n^{(1/3)})}$, n -number of digits.
  - Many cryptosystems based on hardness of factoring.
  - Polynomial time quantum algorithm [Shor, 1994]
  - Similar quantum algorithm solves discrete log.

- **Searching**
  - Find if there exists i for which $x_i=1$.
  - Queries: input i, output $x_i$.
  - Classically, n queries.
  - Quantum, $O(\sqrt{n})$ queries [Grover, 1996].
  - Speeds up exhaustive search.

| $x_1$ | $x_2$ | $x_3$ | | $x_n$ |
|---|---|---|---|---|
| 0 | 1 | 0 | ... | 0 |

# Why Quantum Computation? (other)

- **Key Distribution**
  - Two parties desire to create a secret shared key by using a channel that can be eavesdropped.
  - Classically: secure if discrete log hard.
  - Quantum: secure if quantum mechanics valid [Bennett, Brassard, 1984].
  - No extra assumptions needed

# Why Quantum Computation? (other)

- Simulating Quantum Systems
- Testing Matrix Multiplication
- Element Distinctness
- Graph Problems such as Graph Traversal
- Pell's Equaiton
- … (more applications are being developed)

# Central Question that Remains?

- **Is it actually possible to construct a practical computer that performs calculations on qubits?**

# OUR GOAL

- **Is it actually possible to construct a practical computer that performs calculations on qubits?**
- **We identify the central scalability issues for computationally relevant quantum computers, and describe theoretical microarchitectural abstractions for the physical quantum architecture elements needed to achieve scalability.**

# Outline

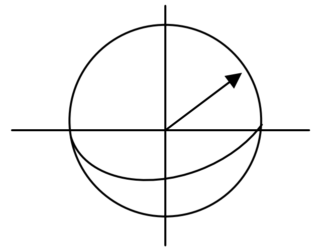
- **How Does Quantum Computing Work?**
- Large-Scale Architecture Requirements
- A Quantum Logic Array Architecture (QLA)
- Specialized Quantum Architecture (CQLA)
- Programming the Architecture
- Related work and other Architecture Models

| Classical Circuits | Quantum Circuits |
|---|---|
| Signal states are bit vectors: X = 0010... | Signal states are vectors expressed as a superposition of binary bit vectors with complex coefficients. |
| Circuit behavior is governed by classical physics. | Circuit behavior is governed by quantum mechanics. |
| No restrictions on measuring and copying signals exist. | Severe restrictions exist on copying and measuring states. |
| Operations are defined by boolean algebra. | Operations are defined by linear algebra and represented as unitary matrices. |
| Implemented with fast, reliable, and scalable CMOS technologies. | Technologies are slow, unreliable, and not trivially scalable. |
| Small set of universal gates exist. | Small set of universal gates exist. |

# Quantum Data (qubits)

- One **qubit** is a single unit of quantum data, represented as a two-element state vector with complex amplitudes $\alpha$ and $\beta$:

$$|\Psi\rangle_1 = \alpha|0\rangle + \beta|1\rangle \rightarrow \begin{bmatrix}\alpha \\ \beta\end{bmatrix}, where \ : |\alpha|^2 + |\beta|^2 = 1$$



- Two **qubits** are simply the combination of two one-qubit vectors:

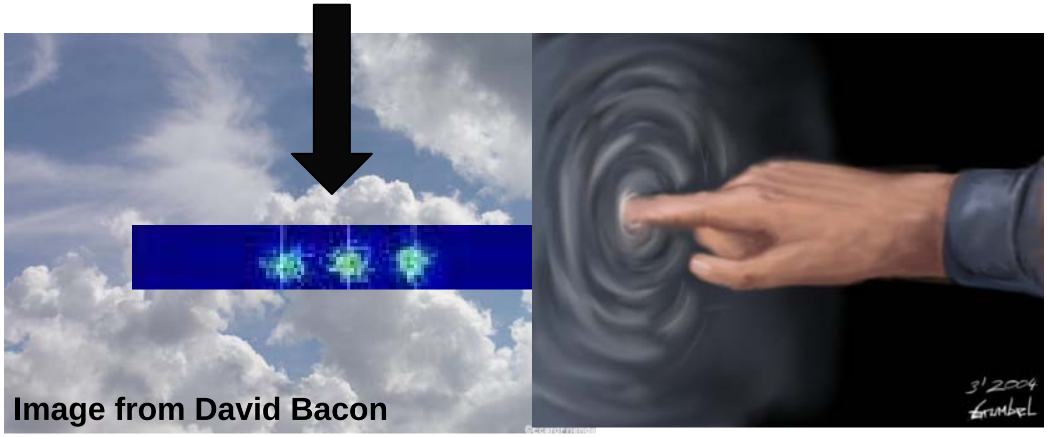
$$|\Psi\rangle_2 = |\Psi\rangle_1 \otimes |\Psi\rangle_1 = c_0|00\rangle + c_1|01\rangle + c_2|10\rangle + c_3|11\rangle$$

- In General:

$$|\Psi\rangle_n = \sum_{i=0}^{2^n-1} c_i |x_i\rangle$$

# Quantum Gates

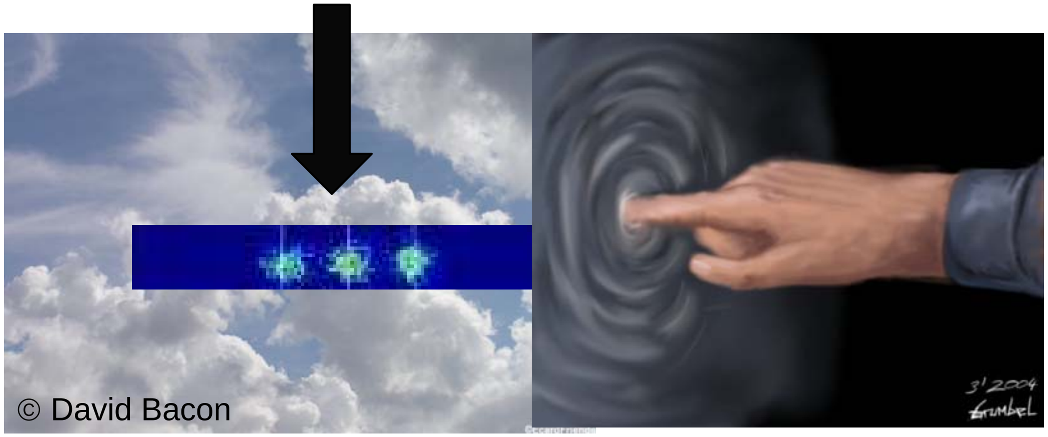
Quantum Registers




environment control

# Quantum Gates

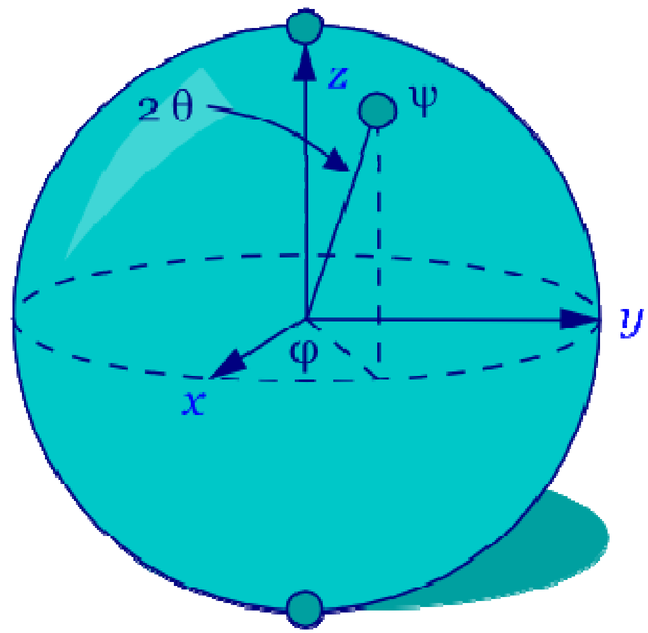
Quantum Registers




environment control

Gates can be defined as a rotation of the qubit state within the space described by the three dimensional Bloch Sphere

$$\left|\Psi\right\rangle_1 = \cos\theta\left|0\right\rangle + e^{i\phi}\sin\theta\left|1\right\rangle$$

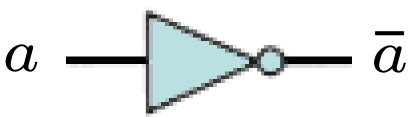


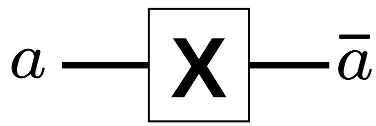
# Quantum Gates

- Represented as unitary matrices that act on the vector describing the system.
- Recall: a system of N qubits is described by a vector of $2^N$ elements.
- A gate acting on all qubits is a $2^N$ x $2^N$ unitary matrix.
- Any unitary 2 x 2 matrix can be written as the general rotation matrix:

$$\begin{pmatrix} e^{i\alpha/2} & 0 \\ 0 & e^{-i\alpha/2} \end{pmatrix} \cdot \begin{pmatrix} \cos\theta/2 & \sin\theta/2 \\ -\sin\theta/2 & \cos\theta/2 \end{pmatrix} \cdot \begin{pmatrix} e^{i\beta/2} & 0 \\ 0 & e^{-i\beta/2} \end{pmatrix}$$

# Quantum Gates: “NOT” Gate



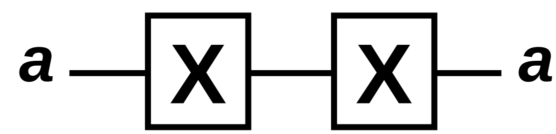

(One Input – One Output)

- **Input state:** $\alpha|0\rangle + \beta|1\rangle$
- **Output state:** $\beta|0\rangle + \alpha|1\rangle$
- **Pure states are mapped thus:** $|0\rangle \rightarrow |1\rangle$ **and** $|1\rangle \rightarrow |0\rangle$
- **Gate operator (matrix) is**

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad |0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

- **As expected:**

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



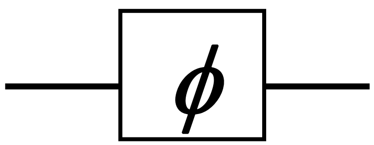
# Quantum Gates: Hadamard and Phase Gates

$$|a\rangle \xrightarrow{\boxed{H}} \frac{|0\rangle + (-1)^a |1\rangle}{\sqrt{2}}$$

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

**maps**

$$\begin{cases} |0\rangle \rightarrow \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right) \\ |1\rangle \rightarrow \frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right) \end{cases}$$

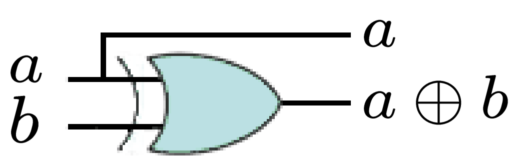


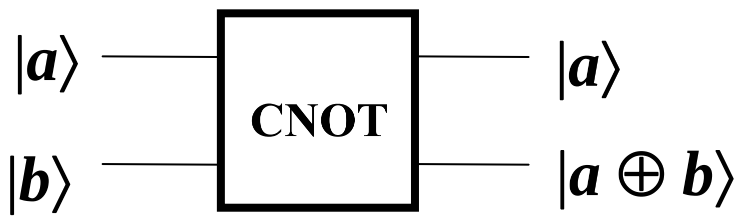

$$\phi = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\phi} \end{pmatrix}$$

**maps**

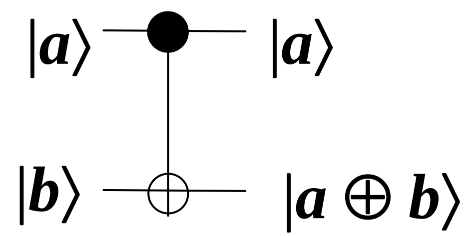
$$\begin{cases} |0\rangle \rightarrow |0\rangle \\ |1\rangle \rightarrow e^{i\phi}|1\rangle \end{cases}$$

# Quantum Gates: Controlled-NOT Gate




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



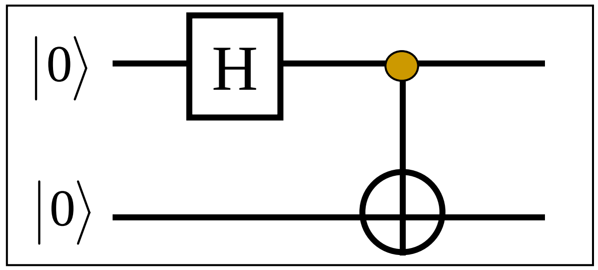

- CNOT maps $|\boldsymbol{a}\rangle|0\rangle \rightarrow |\boldsymbol{a}\rangle|\boldsymbol{a}\rangle$ and $|\boldsymbol{a}\rangle|1\rangle \rightarrow |\boldsymbol{a}\rangle|$|**NOT** $\boldsymbol{a}\rangle$
- $|\boldsymbol{a}\rangle|0\rangle \rightarrow |\boldsymbol{a}\rangle||\boldsymbol{a}\rangle$ looks like cloning, but it's not.
- Combined with single-qubit gates it can be used to build the *Controlled-Controlled-NOT* gate, which is the quantum equivalent of a NAND gate.

# Quantum Circuits

- A quantum circuit is a sequence of quantum gates, linked by “wires”
- The circuit has fixed “width” corresponding to the number of qubits being processed
- Logic design (classical and quantum) attempts to find circuit structures for needed operations that are
  - Functionally correct
  - Independent of physical technology
  - Optimized for some suitable cost function
- Quantum logic design is not well developed!

# Quantum Circuits: EPR Pair

$|00\rangle$ ⇨

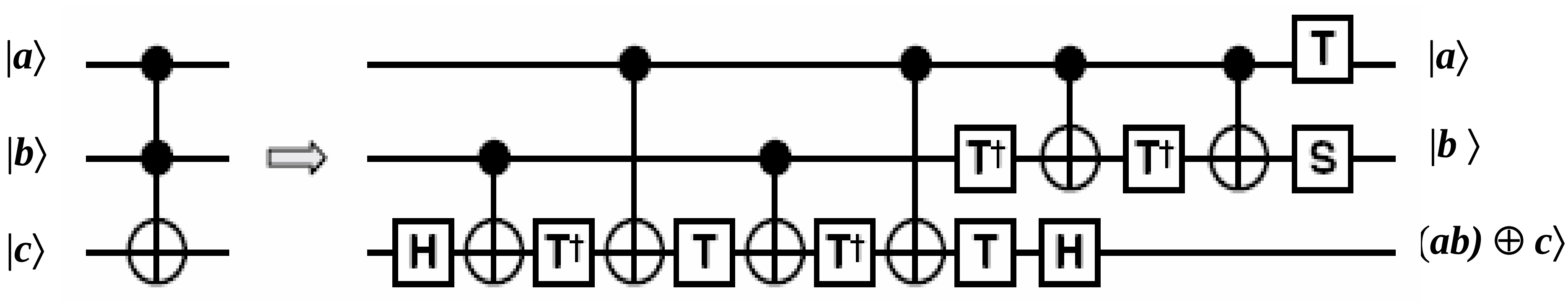


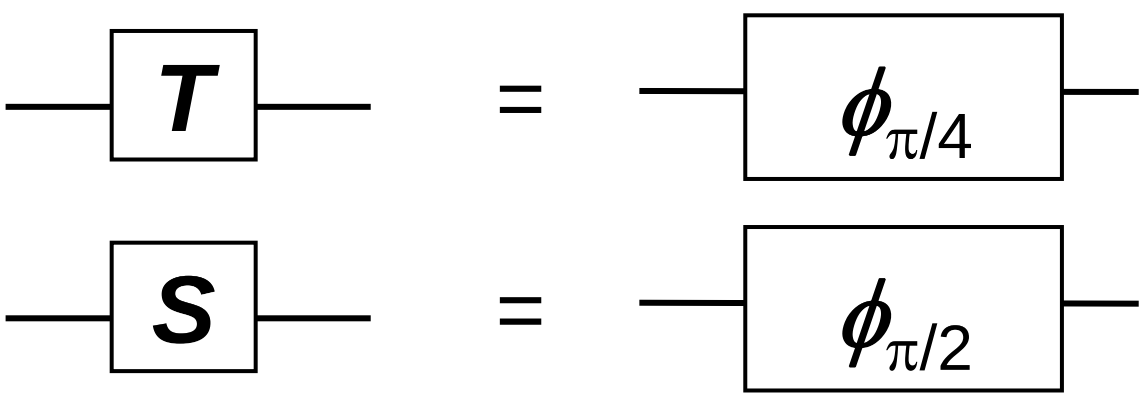

⇨ $\frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle$

$$|00\rangle \rightarrow \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|10\rangle \rightarrow \boxed{\frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle}$$

Pure Entangled State (EPR Pair) used for Teleportation

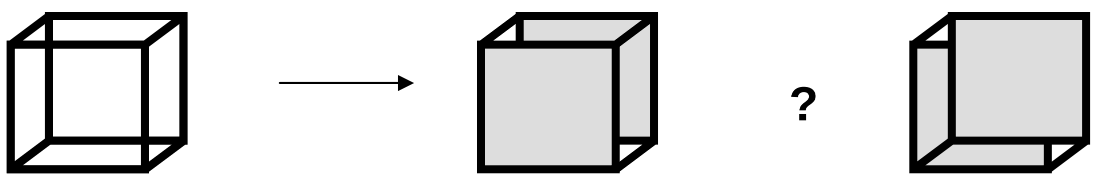
# Quantum Toffoli Gate



# Quantum Measurement

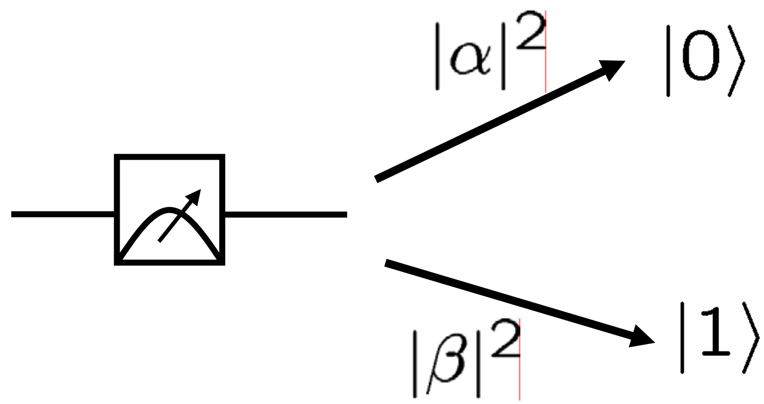
- From all possible $2^N$ states in a collection of N qubits, measurement yields a single N-bit bitstring corresponding to the probability amplitude of that state.
- It is destructive since it makes the resulting bitstring the new state of the measured group of qubits. Measurement is irreversible.



# Quantum Measurement

- From all possible $2^N$ states in a collection of N qubits, measurement yields a single N-bit bitstring corresponding to the probability amplitude of that state.
- It is destructive since it makes the resulting bitstring the new state of the measured group of qubits. Measurement is irreversible.

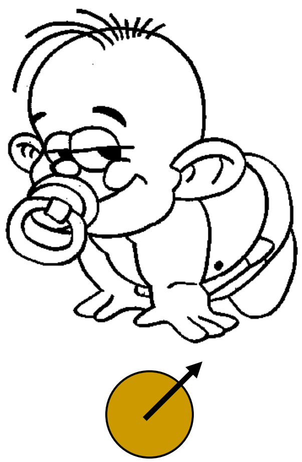
$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle$$



# Quantum Teleportation (Bennett 93')

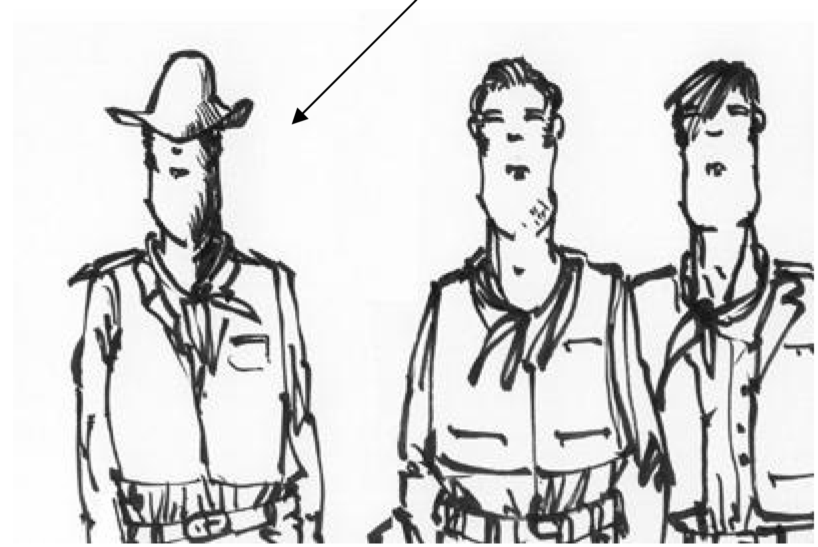
# Quantum Teleportation

**ALICE**

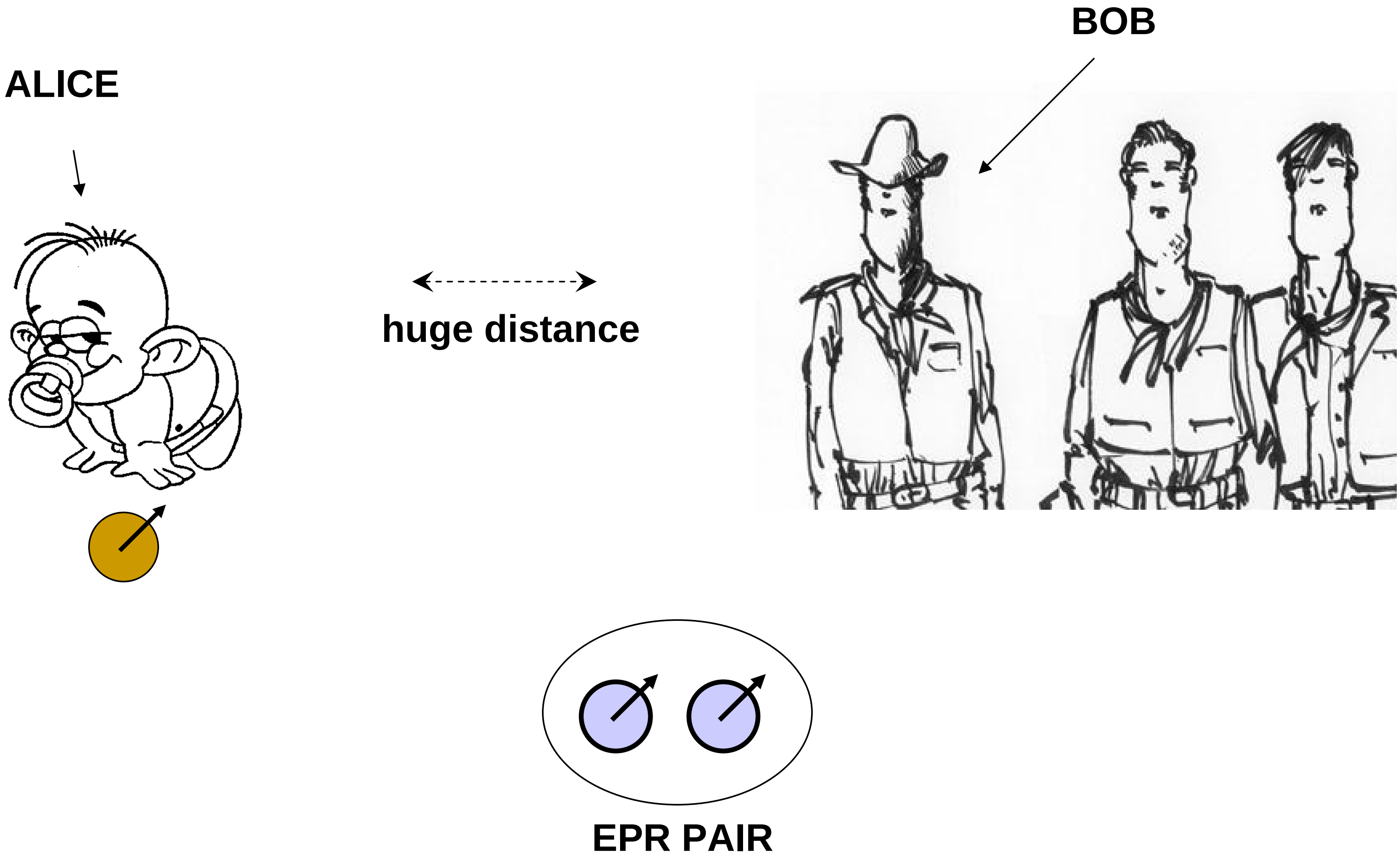


**some separation**

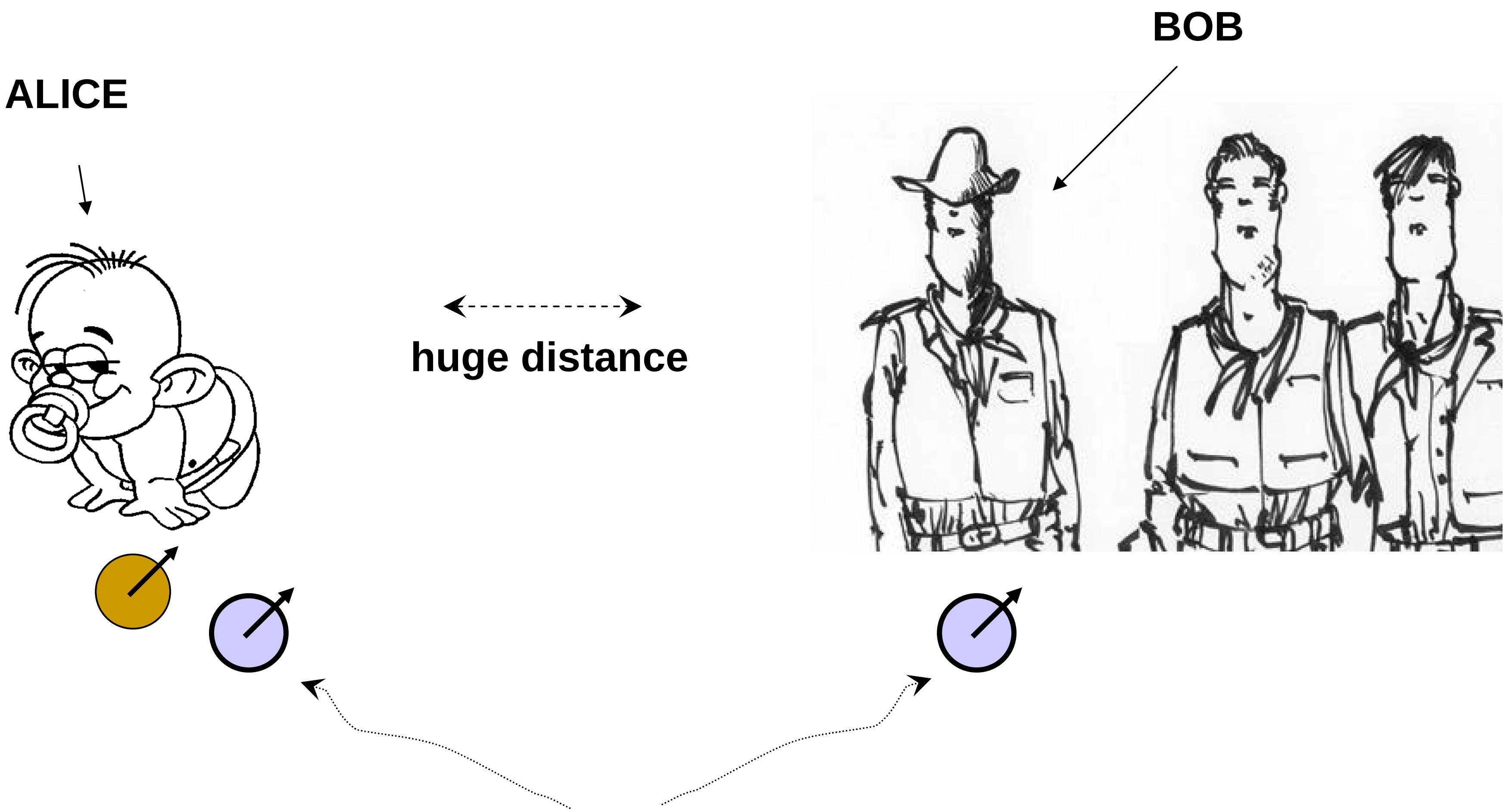
**BOB**



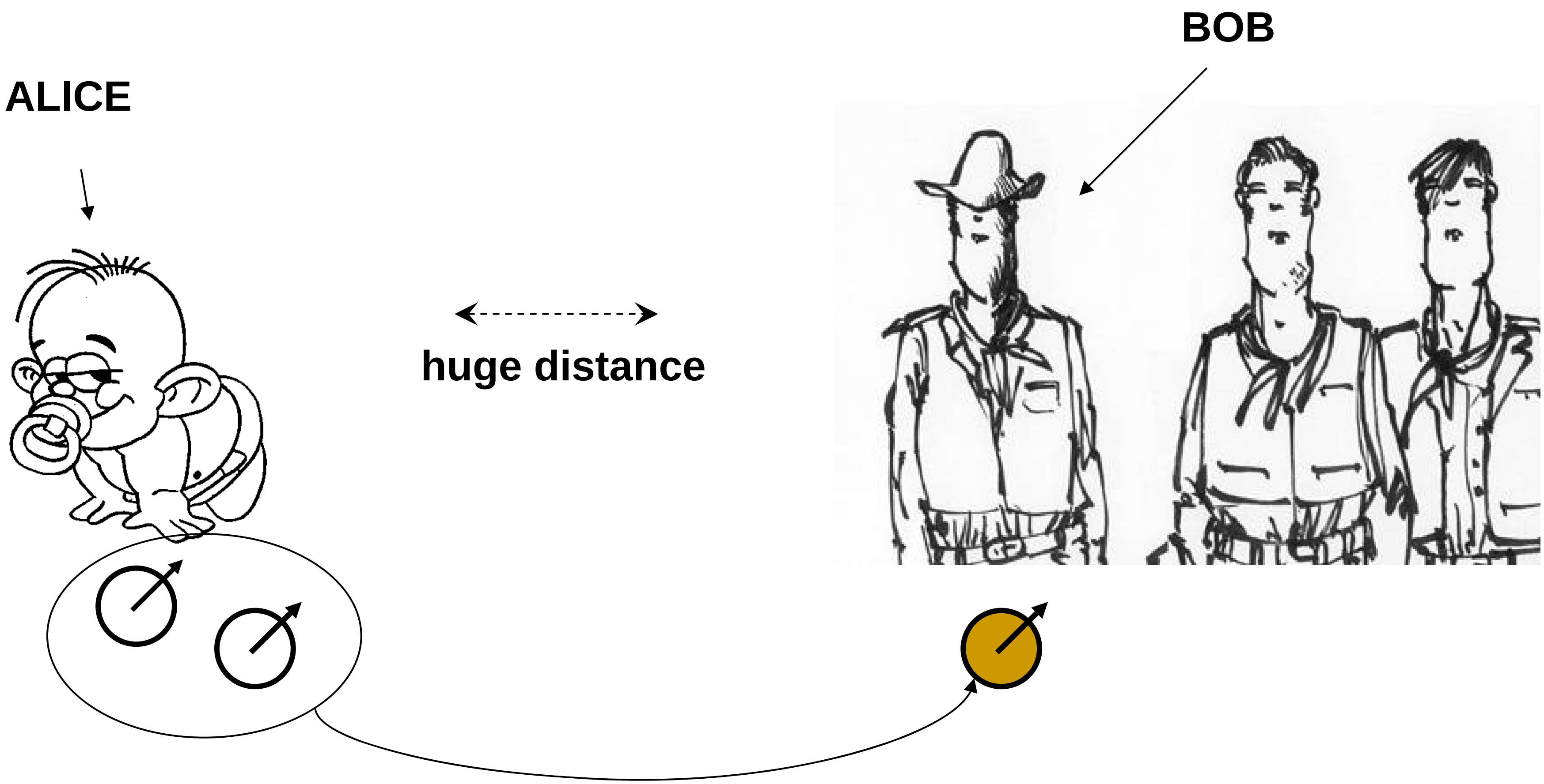
# Quantum Teleportation



# Quantum Teleportation

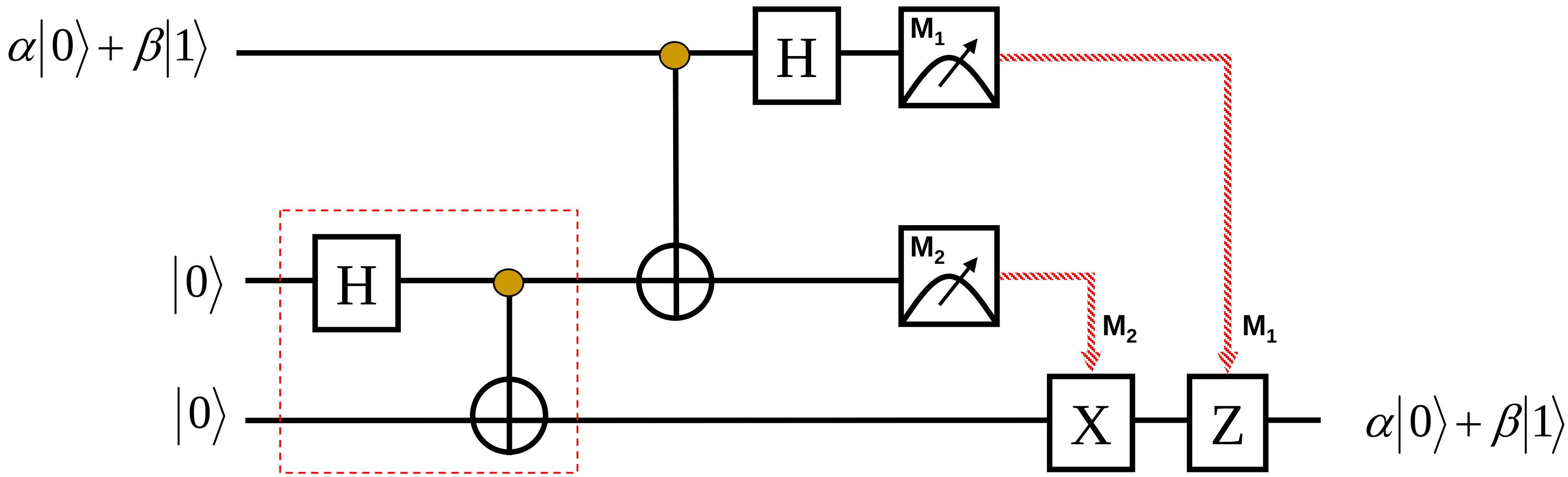


# Quantum Teleportation

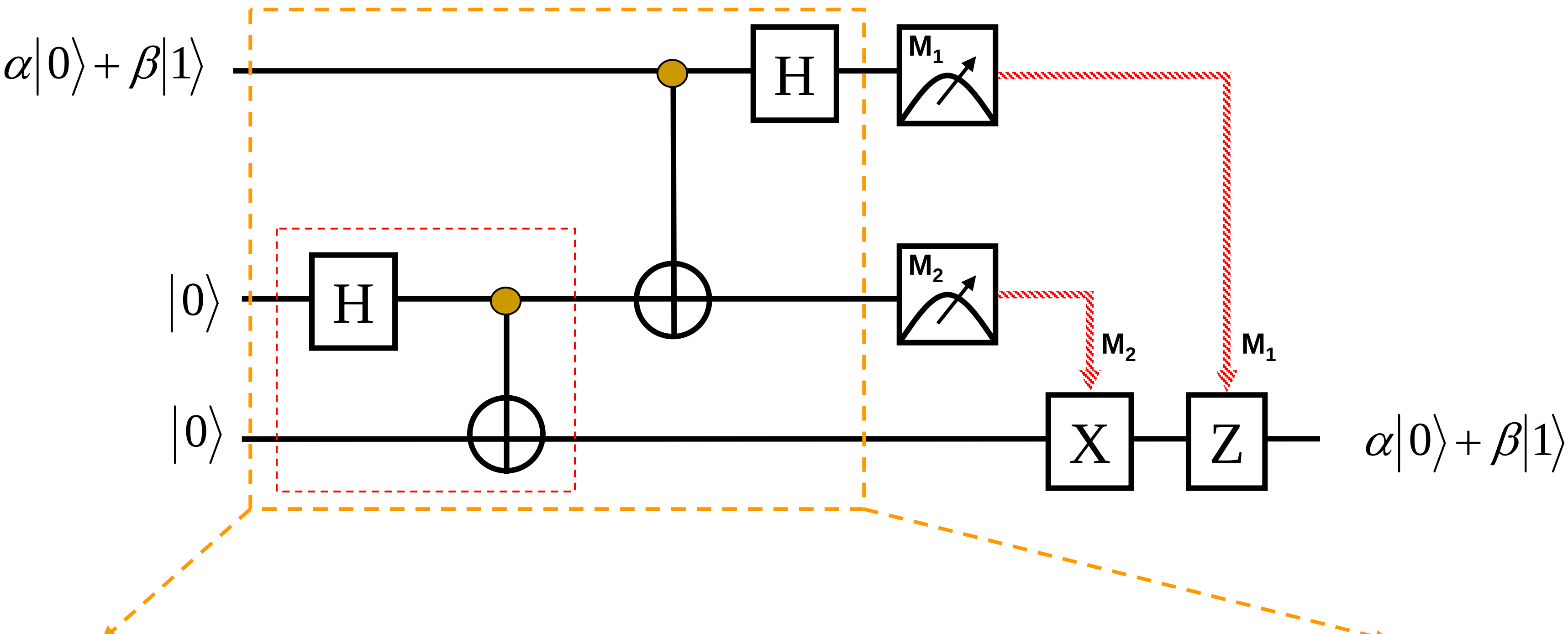



**measurement result is used to recreate Alice’s qubit at Bob’s Location**

# Understanding Teleportation

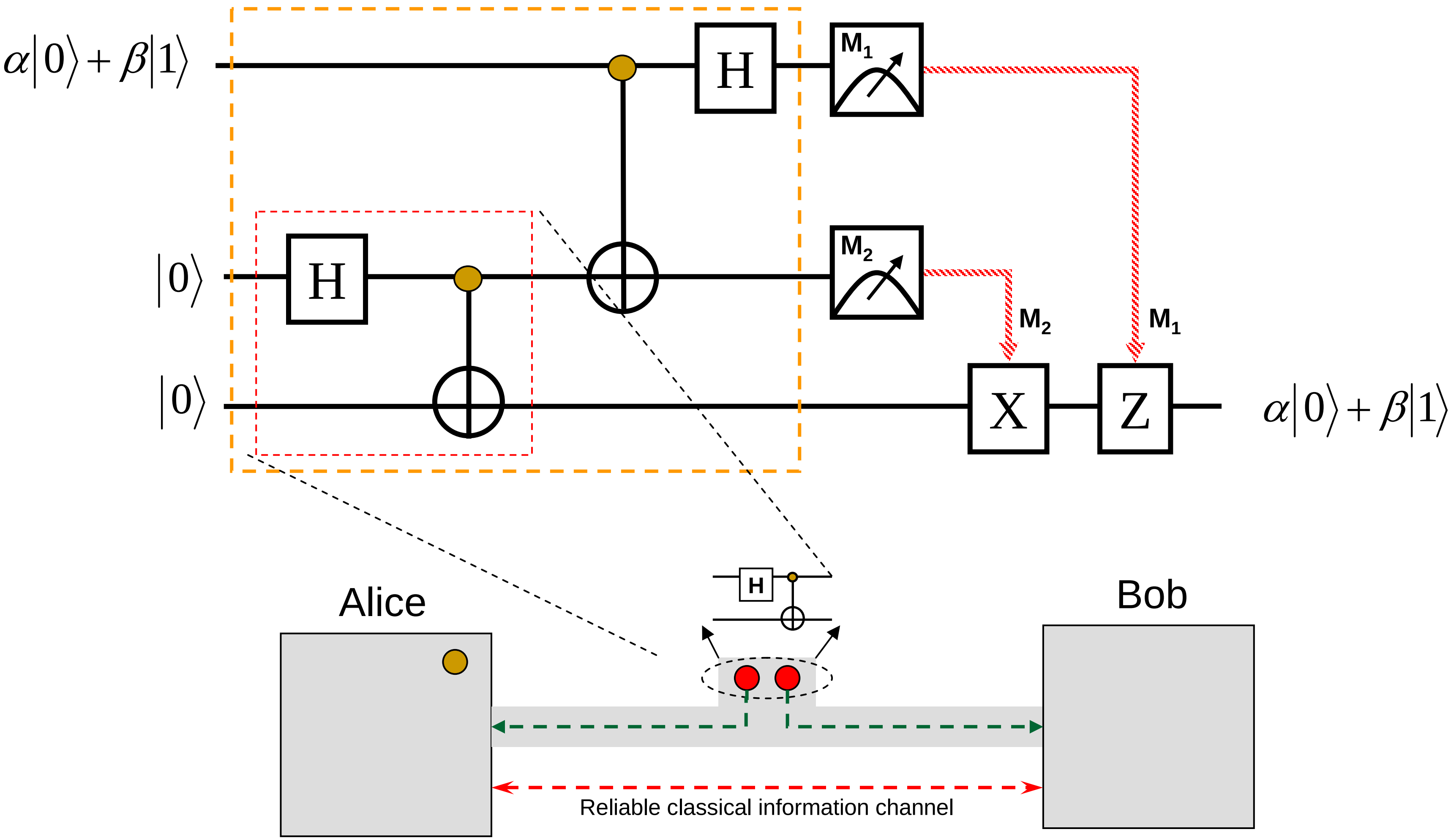


# Understanding Teleportation (Cont.)




$$|00\rangle\underbrace{\left(\alpha|0\rangle + \beta|1\rangle\right)} + |01\rangle\underbrace{\left(\alpha|1\rangle + \beta|0\rangle\right)} + |10\rangle\underbrace{\left(\alpha|0\rangle - \beta|1\rangle\right)} + |11\rangle\underbrace{\left(\alpha|1\rangle - \beta|0\rangle\right)}$$

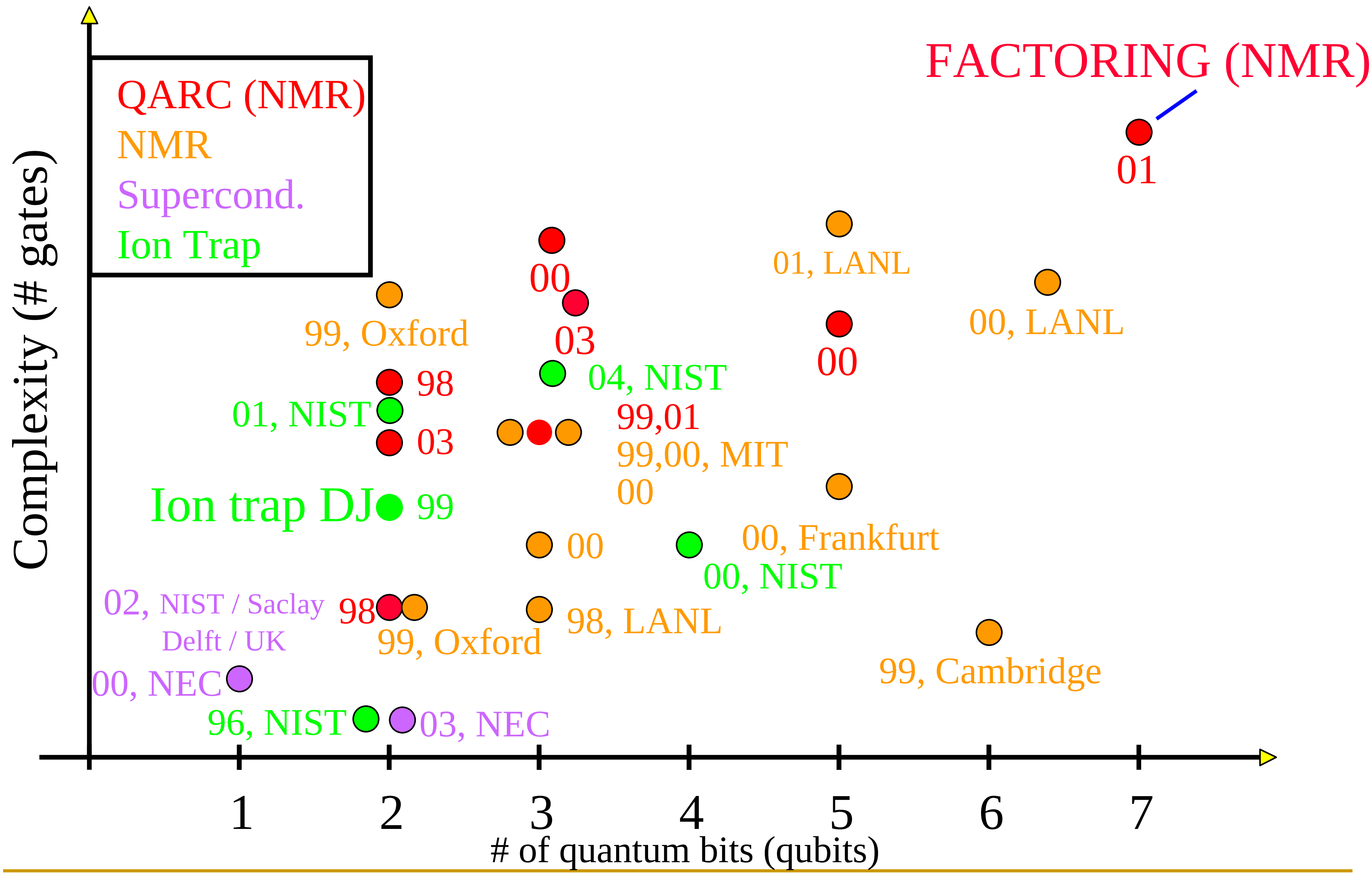
# Understanding Teleportation (Cont.)



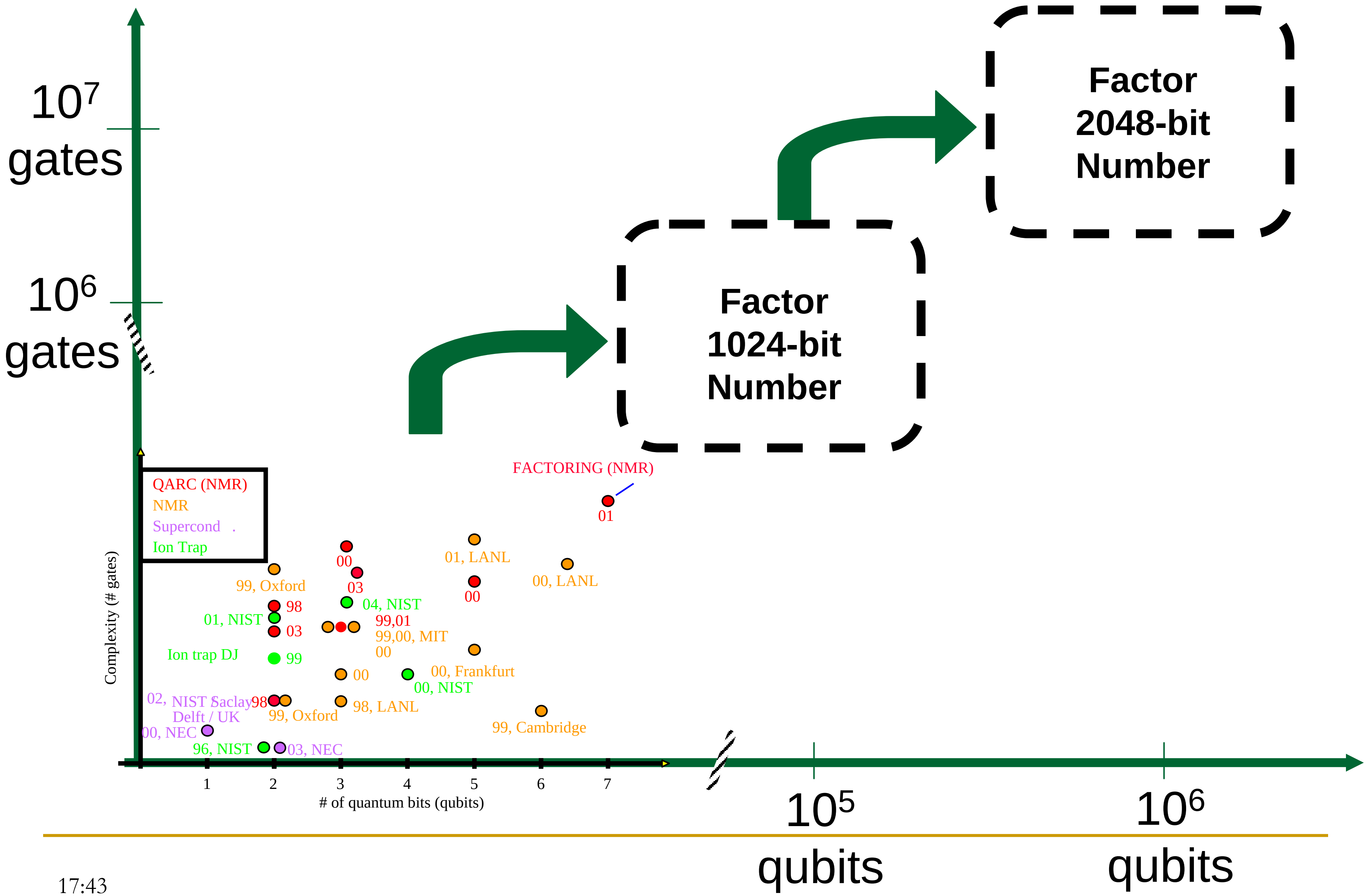
# Outline

- How Does Quantum Computing Work?
- **Large-Scale Architecture Requirements**
- A Quantum Logic Array Architecture (QLA)
- Specialized Quantum Architecture (CQLA)
- Programming the Architecture
- Related work and other Architecture Models

# Quantum Computers Today



# Our Goal …



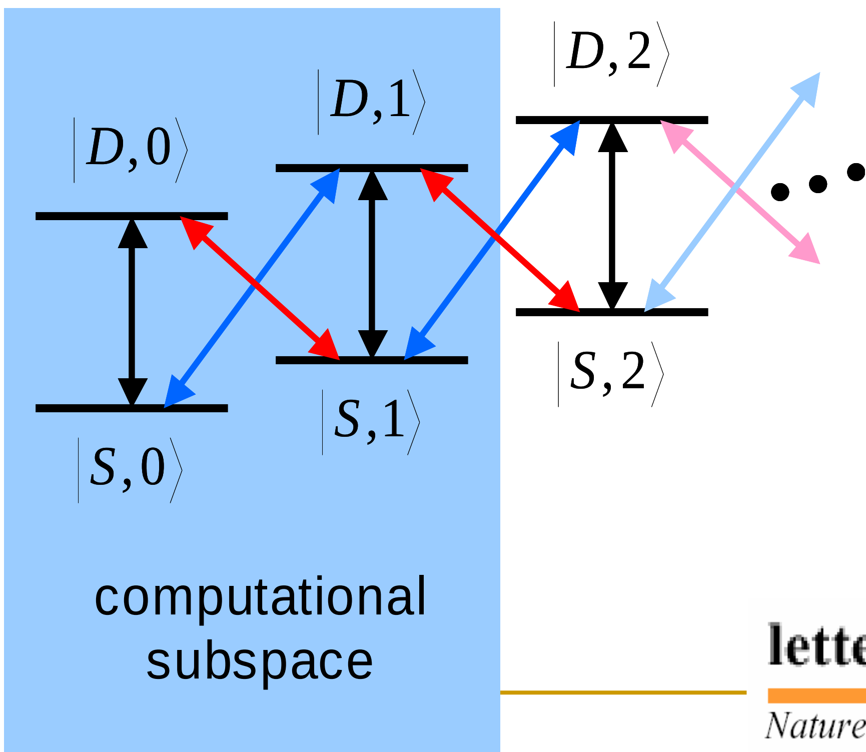
$10^7$ gates
$10^6$ gates
Factor 2048-bit Number
Factor 1024-bit Number
FACTORING (NMR)
QARC (NMR)
NMR
Supercond
Ion Trap
Complexity (# gates)
# of quantum bits (qubits)
1 2 3 4 5 6 7
$10^5$ qubits
$10^6$ qubits

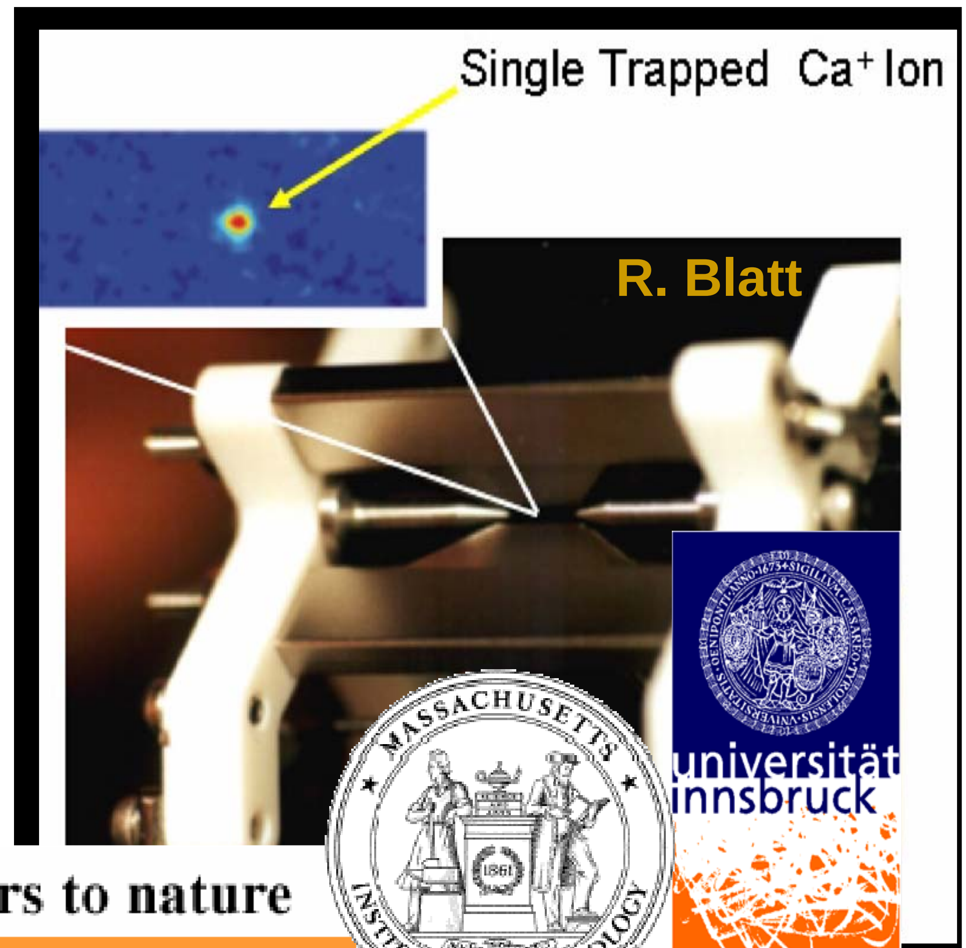
# Building a Quantum Architecture

- Reliable and Realistic Technology
  - Reliable initialization of each qubit
  - Universal set of quantum operations
  - Ability to Measure the system
- Fault-Tolerant Layout and Error Correction
- Efficient Quantum Resource Distributions.

# Ion-Trap Technology

- **Complex quantum control accomplished**
- **Two qubits, ~10 ops**

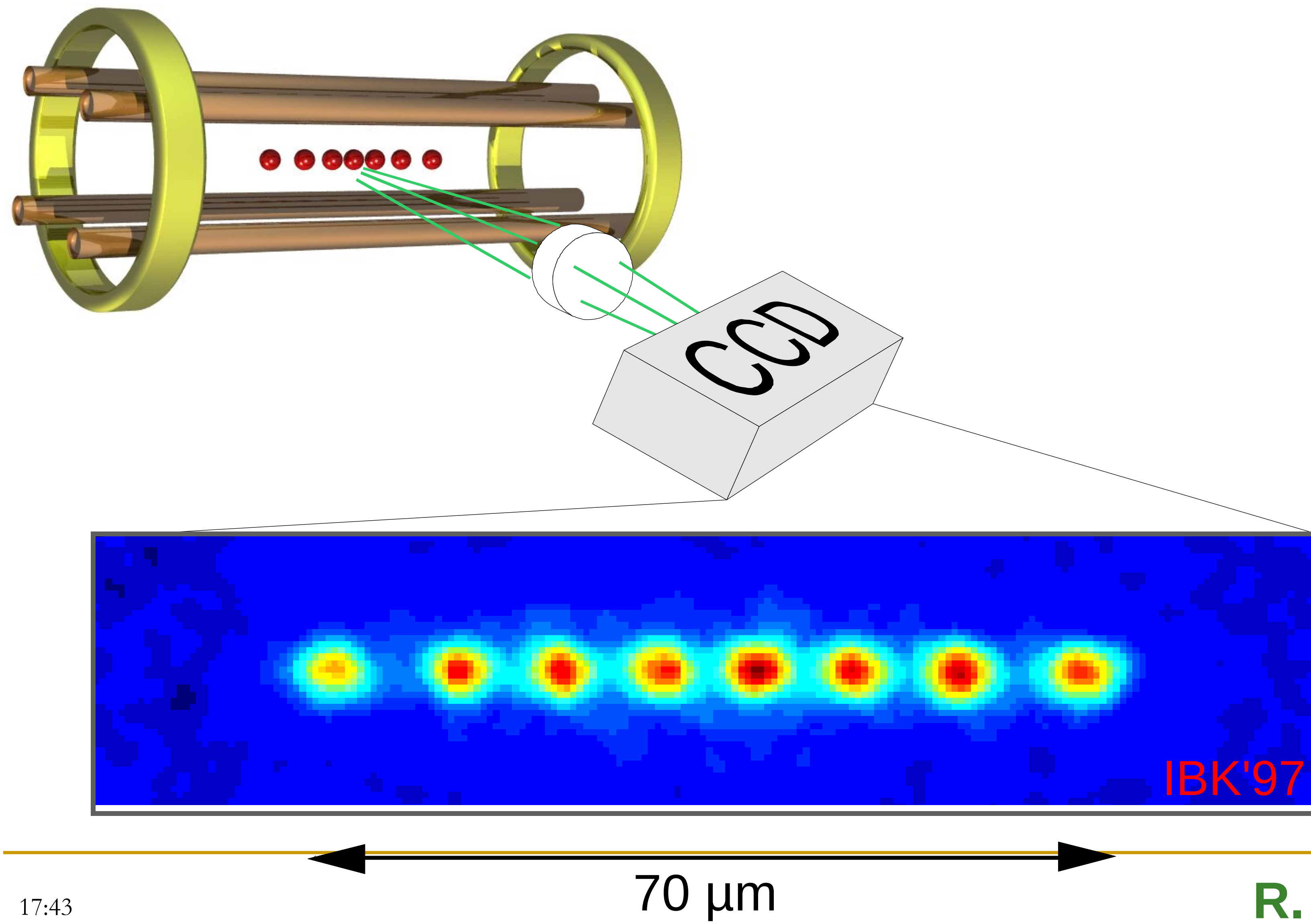



**letters to nature**

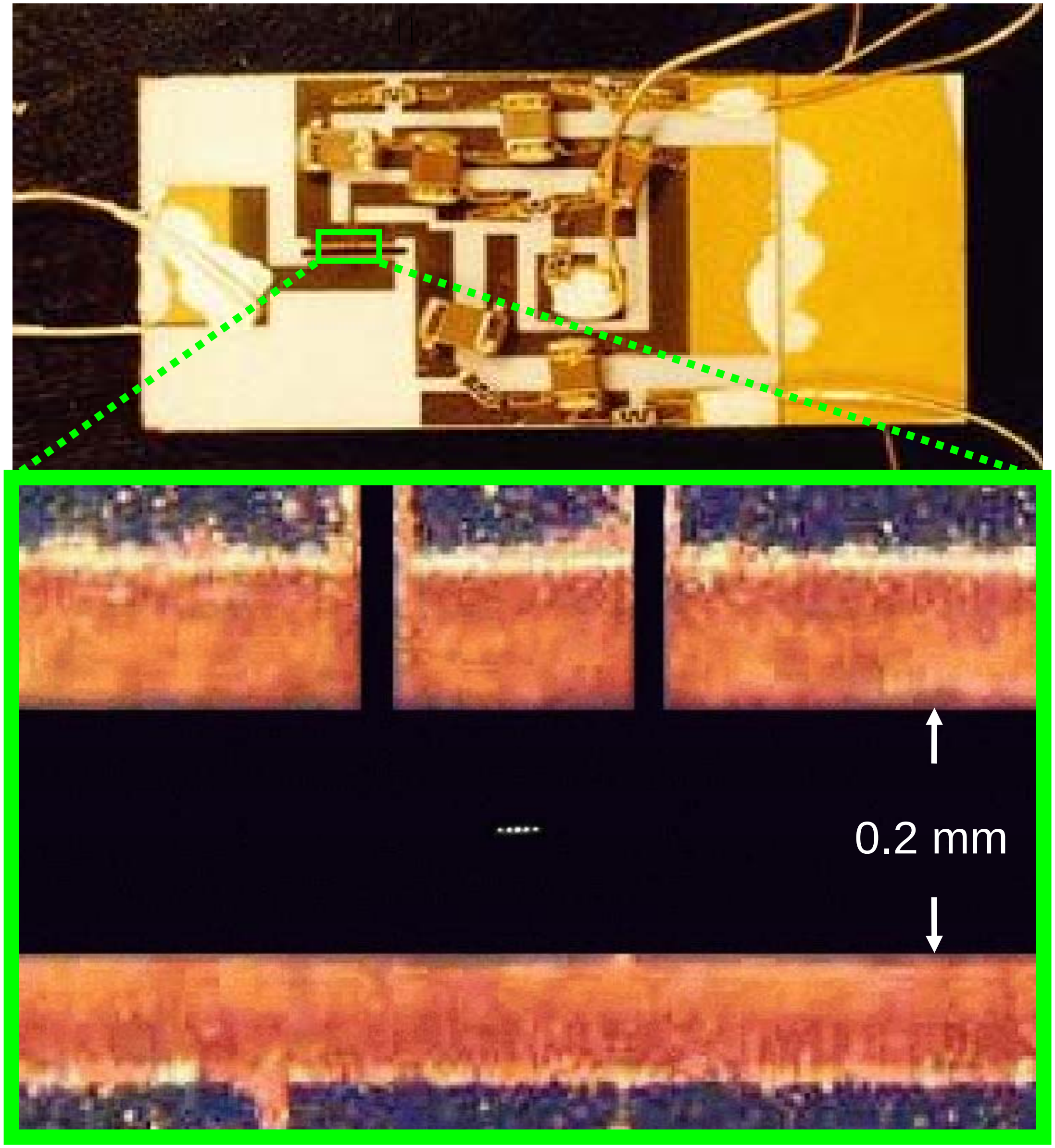
*Nature* **421**, 48 - 50 (2003)

# Trapped-Ion QC

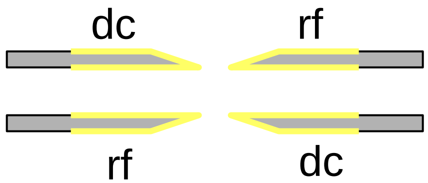


# NIST Boulder: Ion Trap design

( D. Wineland / courtesy D. Liebfried, 2002 )



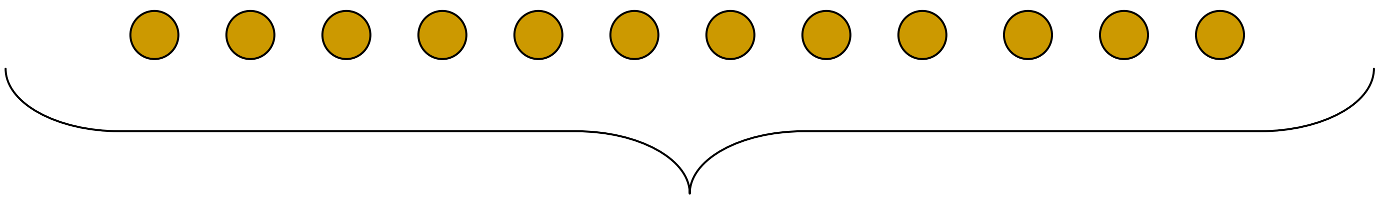

2 wafers of alumina (0.2 mm thick)
gold conducting surfaces (3 μm)
filter electronic on board (SMD)




small trap electrode dimensions

# Major Problem !!!

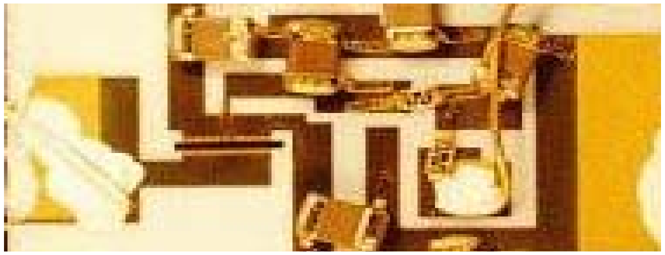
- Quantum properties break down for systems of many qubits. It becomes exponentially difficult to distinguish individual qubit states.



**Large Chain of Ions**

# Scalable Ion-Traps

- Scaling: mictrotraps



(Wineland/NIST)

- **Large-scale QC?**
  - **Teleportation can be used for wiring & code conversion**
  - **Gate errors ~ O($10^{-4}$) possible**

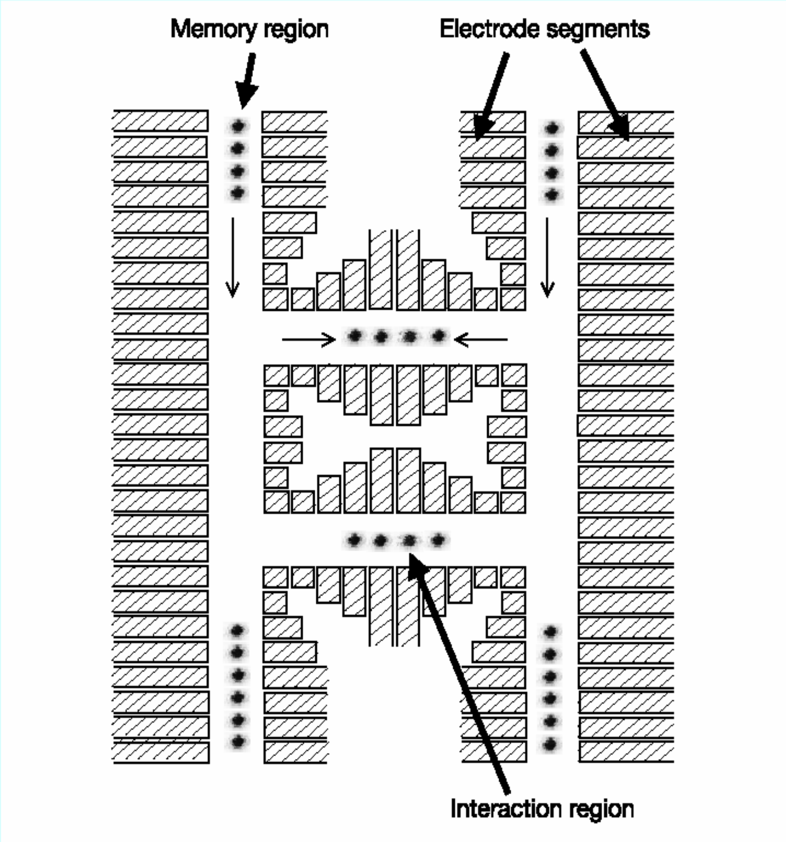
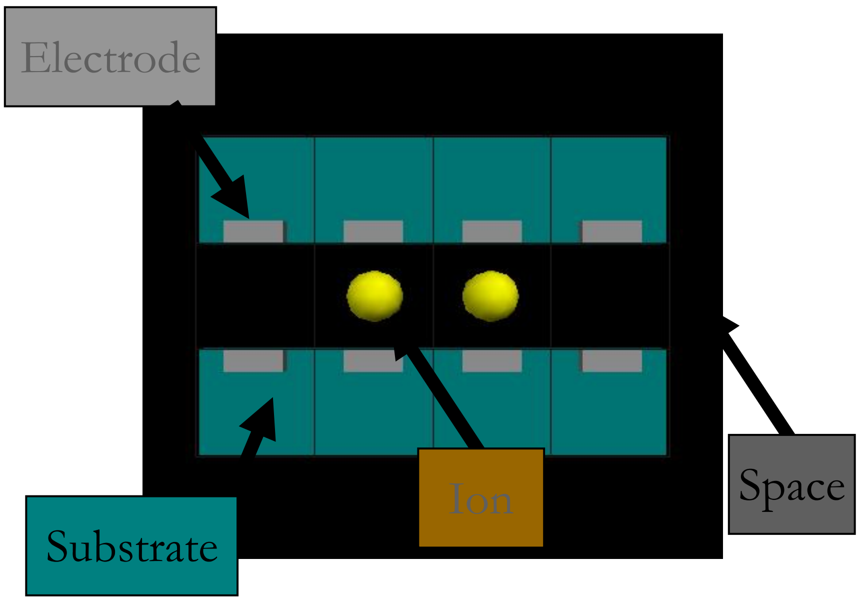



**Figure 1** Diagram of the quantum charge-coupled device (QCCD). Ions are stored in the memory region and moved to the interaction region for logic operations. Thin arrows show transport and confinement along the local trap axis.

**Kielpinski et al, Nature v417, p 709, 2002**

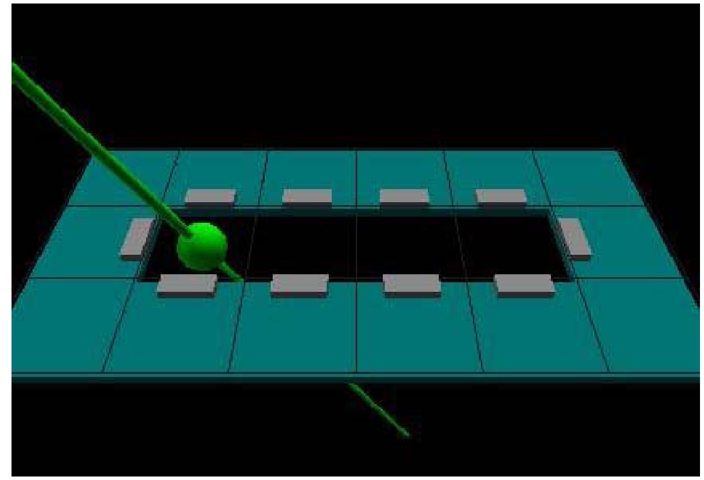
# Ion trap essentials:

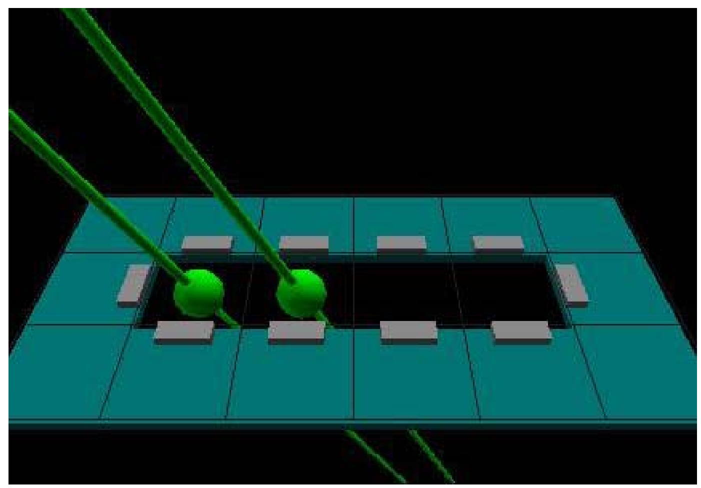



- RF Paul Trap Segments
  - Substrates with attached electrodes for ion trapping and control
- Ions in linear chains
  - Qubits are hyperfine states
  - Qubits are coupled through collective vibrations
- Lasers implement logic gates and measurement

# Gates and Measurement:

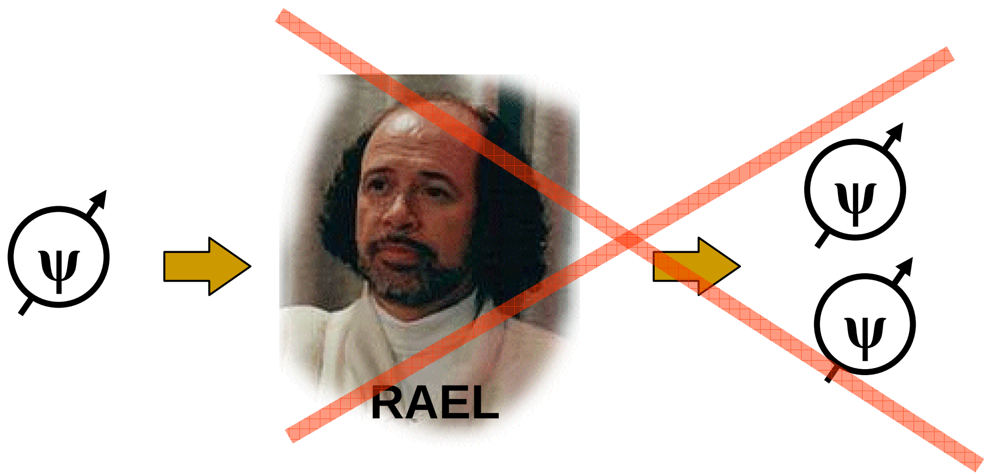
- Single Qubit Gate
  - 1 $\mu$s execution time
  - $P_{fail} = 0.0001$
- Two Qubit Gate
  - 10 $\mu$s execution time
  - $P_{fail} = 0.03$
- Measurement
  - 100 $\mu$s execution time
  - $P_{fail} = 0.01$





# Data Distribution

**Quantum Data cannot be copied**



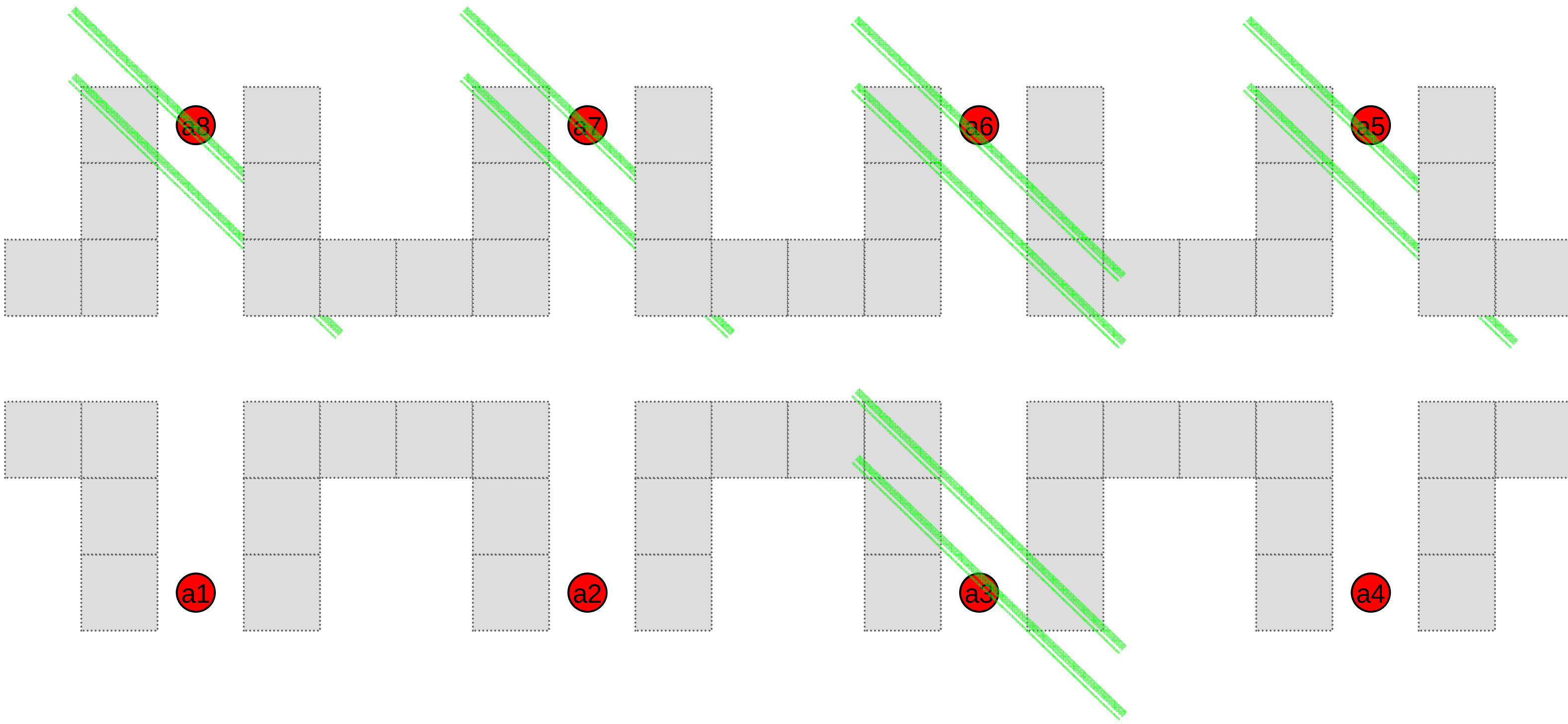

- The state of the qubit must be physically transferred to each new location without leaving a trace in it's current location.

# Data Distribution

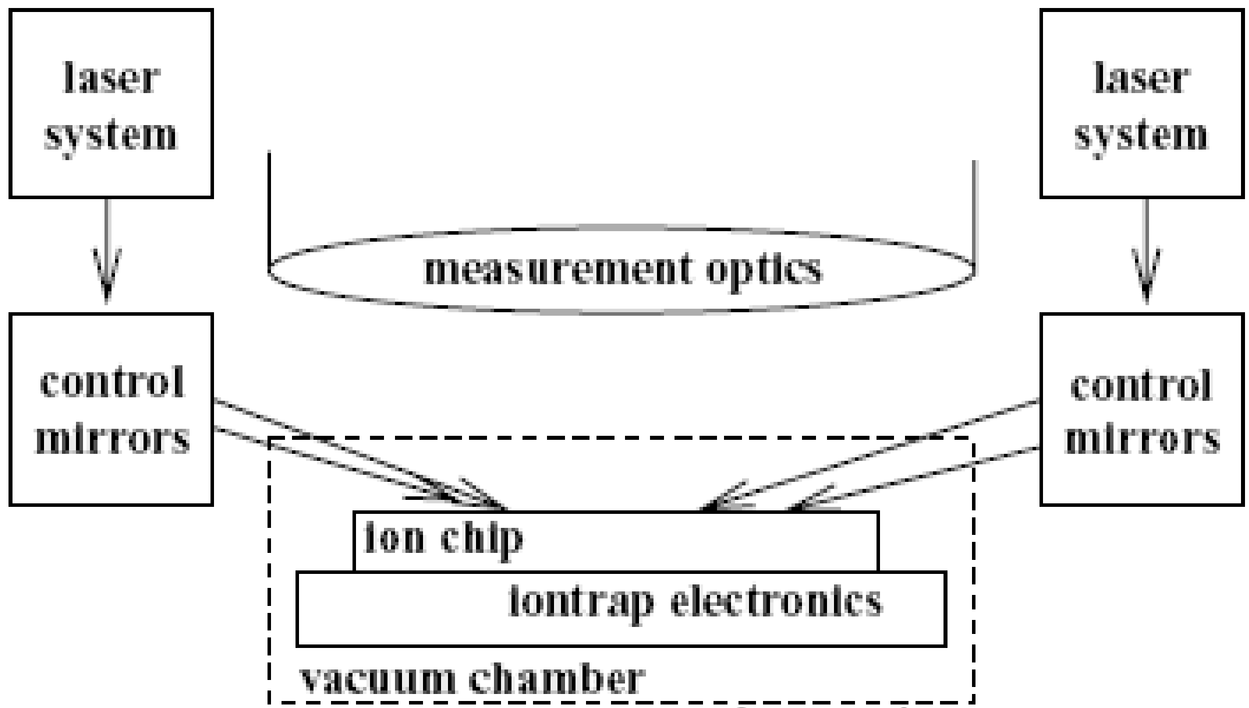
- A good physical qubit implementation is one that allows the qubit to be:
  - protected from the environment while being transported
  - exposed to the environment for quantum logic
  - stationary enough for reliable quantum logic
  - mobile enough for reliable communication

# Data Distribution (Physical Ion Movement)

cnot a1,a8
cnot a6,a2
cnot a7,a2
cnot a3,a6
cnot a3,a5
cnt a4,a5
cnot a2,a1



# Possible Ion-Trap Computer




Steane 04'

# Building a Quantum Architecture

- Reliable and Realistic Technology
  - Reliable initialization of each qubit
  - Universal set of quantum operations
  - Ability to Measure the system
- **Fault-Tolerant Layout and Error Correction**
- Efficient Quantum Resource Distributions.

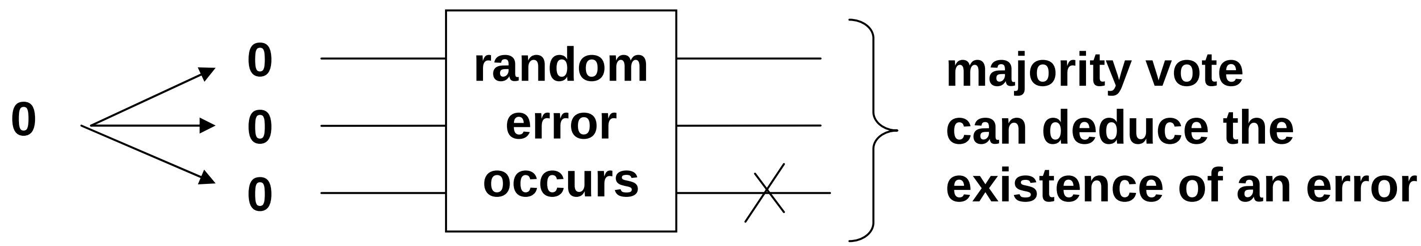
# Fault-Tolerant Quantum Computation

Arbitrary Reliable computers can be constructed from faulty components

- A circuit containing N (error-free) gates can be simulated with probability of error at most $\varepsilon$, using N $\log(N/\varepsilon)$ faulty gates, which fail with probability p, so long as $p<p_{th}$. von Neumann (1956)

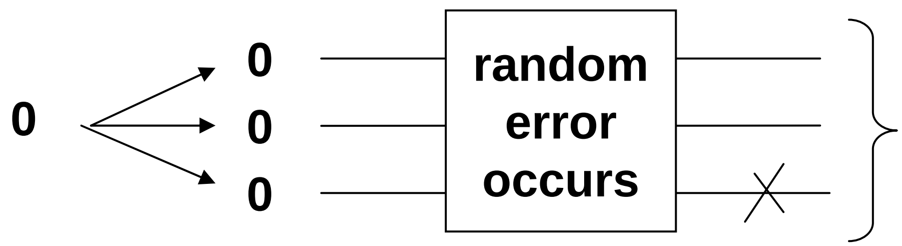
Quantum version: Preskill, Shor, Aharonov, Ben-Or, Gottesman, Zurek, …

# Concept of Redundancy (Shannon 1948)




$$p_{fail} = 3(1-p)p^2 + p^3 \text{ which is} < p \text{ if } p < (1/2)$$

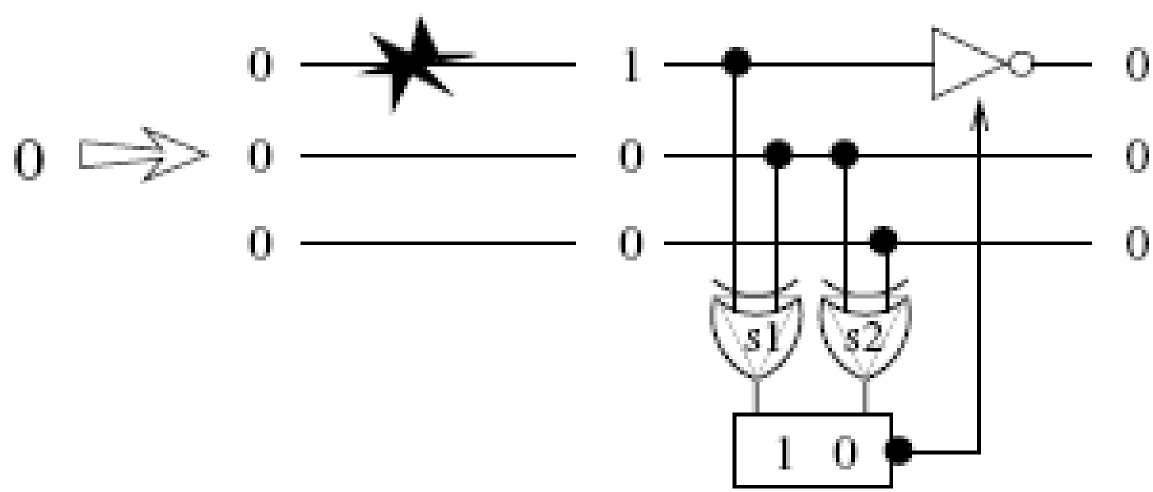
# Concept of Redundancy (Shannon 1948)




**majority vote can deduce the existence of an error**

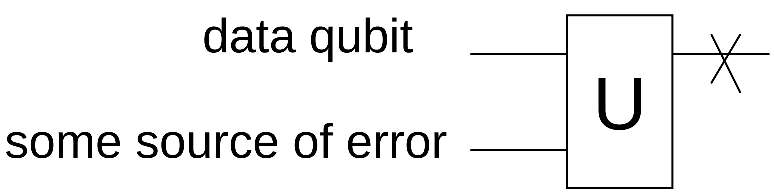
$$p_{fail} = 3(1-p)p^2 + p^3 \text{ which is } < p \text{ if } p < (1/2)$$

**3-Bit Repetition Code can even correct errors.**



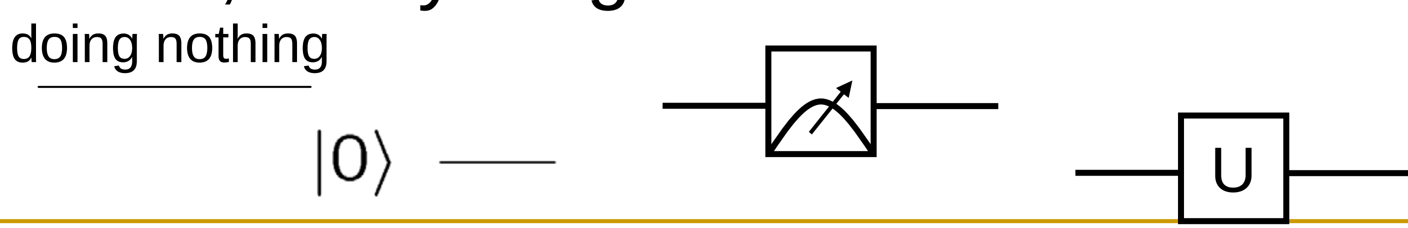
# Obstacles for Quantum Error Correction

- If measurement destroys the state, how can we detect the errors without measuring?
- If quantum data cannot be cloned, how do we invoke the concept of redundancy?
- Number of errors is seemingly infinite:




the error can be an arbitrary gate U

- Moreover, everything fails:



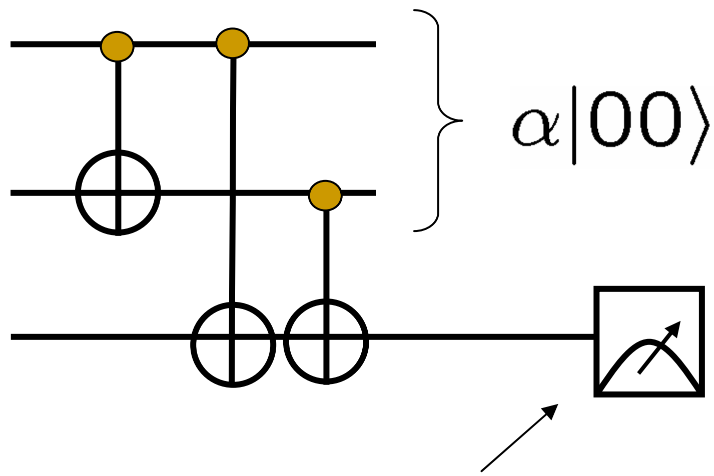
# Solutions

We can measure the error syndrome without destroying the quantum information:

$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

$$|0\rangle$$

$$|0\rangle$$



$$\alpha|00\rangle + \beta|11\rangle$$

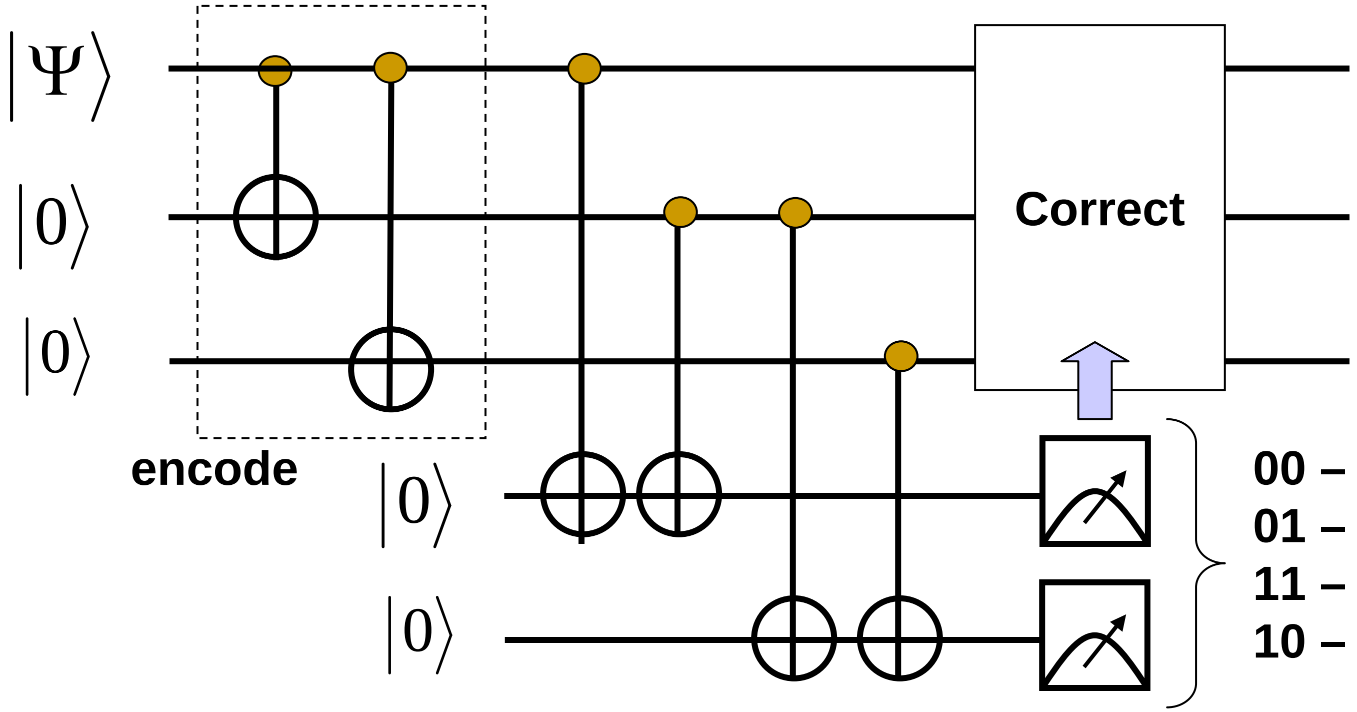
Measurement will return ‘0’ if no bit-flip errors, and ‘1’ if there are bit-flip errors.

By encoding a quantum state with ancillary qubits initialized to $|0\rangle$ we can transfer the error onto additional ancillary qubits which can be measured.

# … Solutions

Similarly we can encode into 3 qubits to correct a bit-flip error on any of the three qubits

$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle \longrightarrow |\Psi\rangle = \alpha|000\rangle + \beta|111\rangle$$



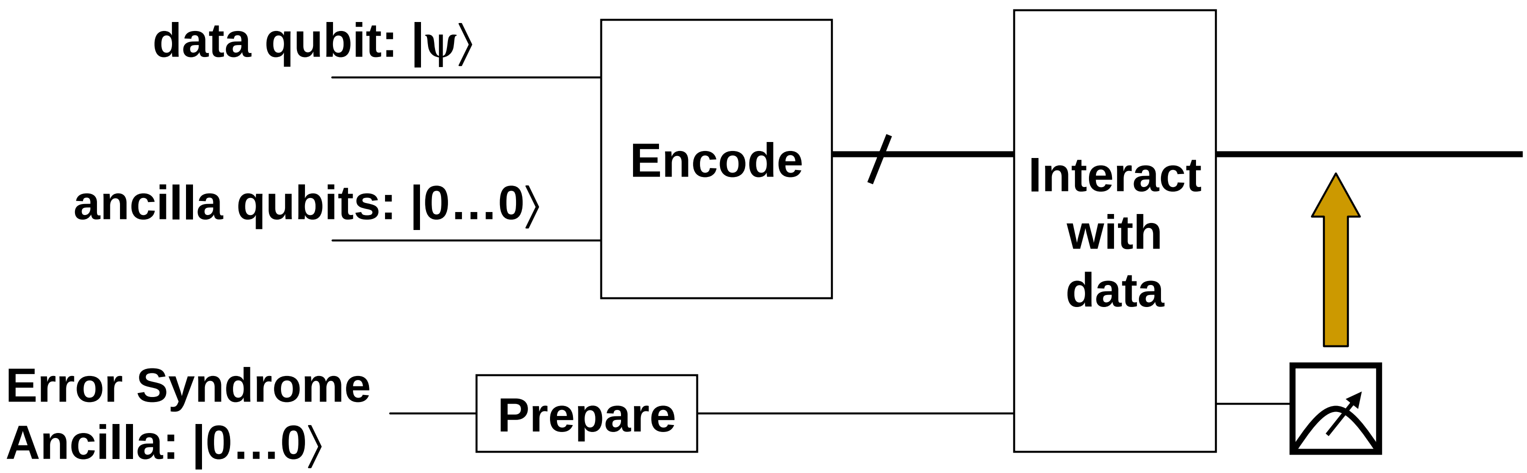

$$\left.\begin{array}{l}\\ \\ \\ \end{array}\right\}$$

**00 – no error**
**01 – error on qubit 3**
**11 – error on qubit 2**
**10 – error on qubit 1**

# … Solutions

- But what about other error types?

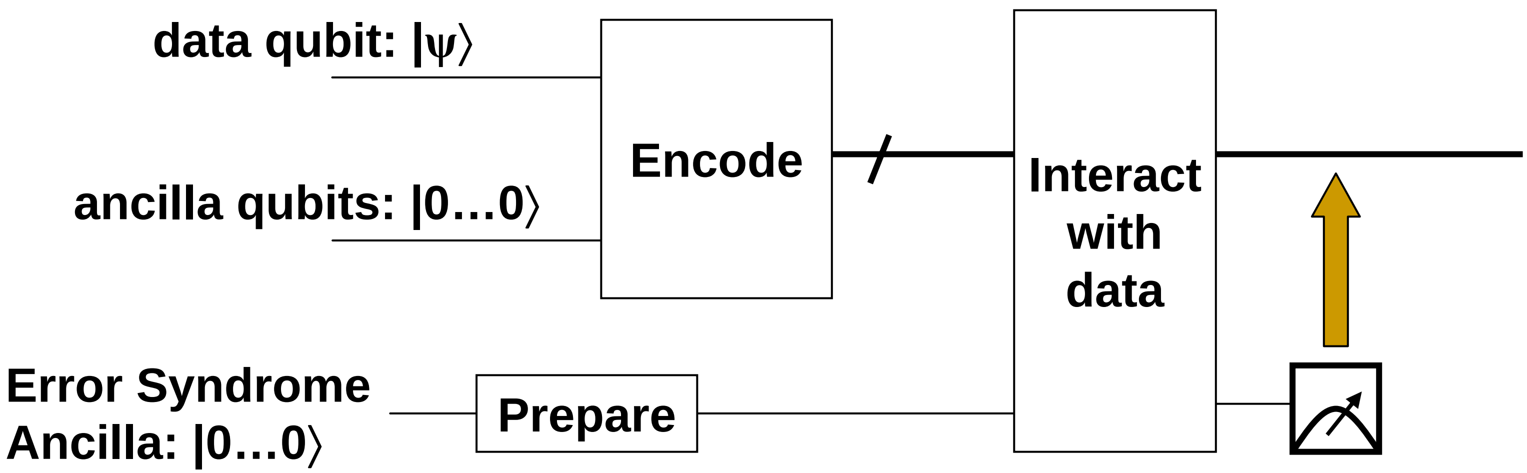
Consider the Generic
Quantum Error Correction Structure:




**Remarkably, all errors in the data can be represented as a combination of X and Z errors whose information is transferred to the syndrome ancilla.**

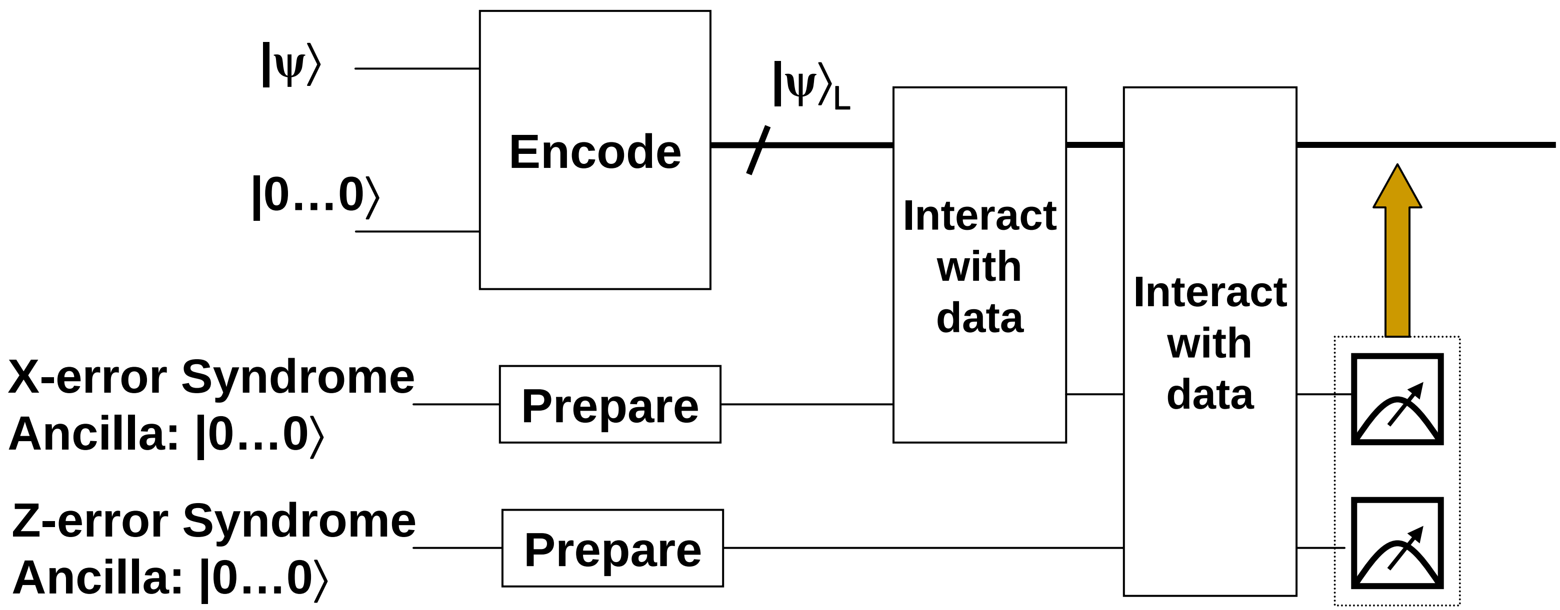
# … Solutions

- The syndrome extraction repeats for both X and Z errors known as the Steane Error Correction Method:

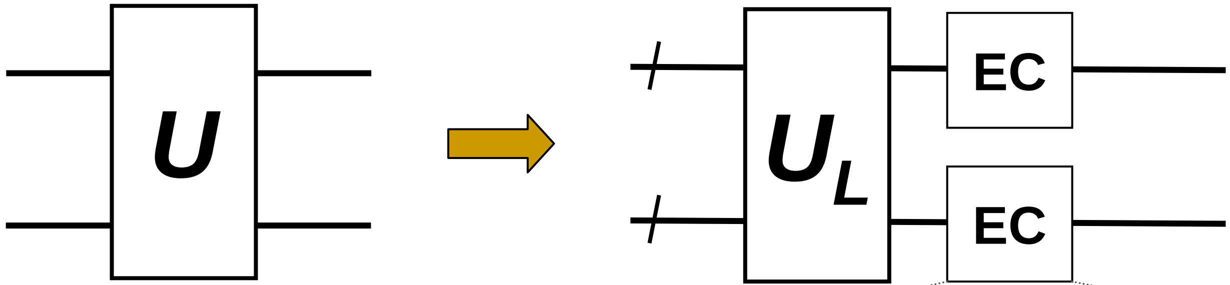


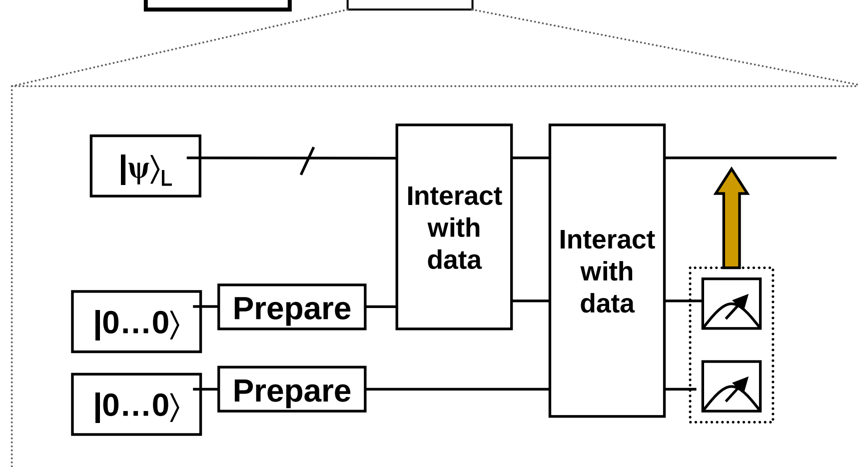

**Remarkably, all errors in the data can be represented as a combination of X and Z errors whose information is transferred to the syndrome ancilla.**

# Steane Method for Error Correction



# Quantum Computation on Logical Qubits



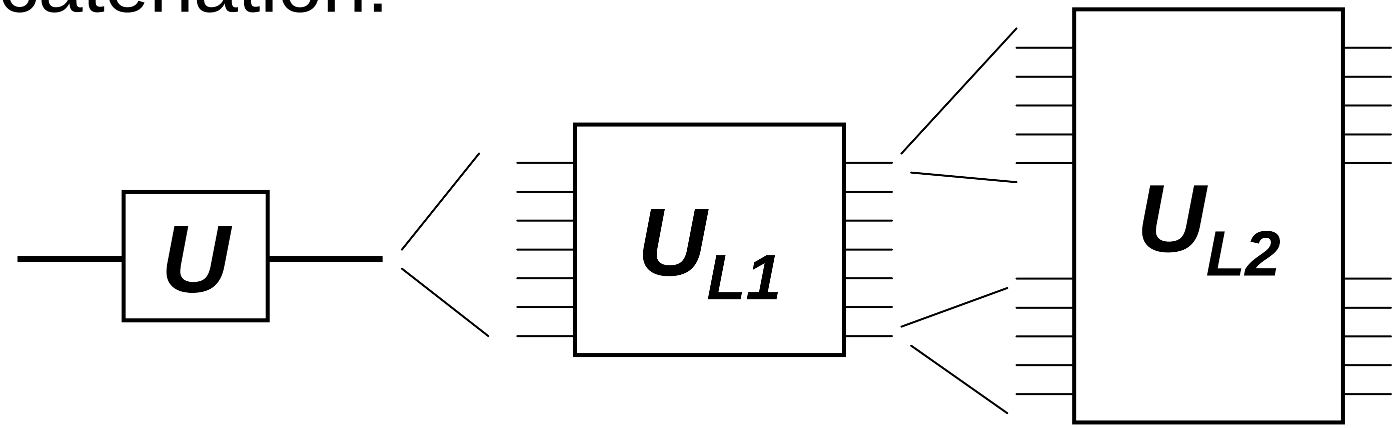

If the error correction code corrects "t" errors, then the failure rate of the logical gate as a whole is:

$$p_{fail} = Ap^{(t+1)}$$

**Where: $p_{fail} < p$ if $p \leq 1/A$**

# Concatenation

- We can achieve arbitrary reliability for logical gates if we continue to increase the level of concatenation:




failure probability $p$ $Ap^2$ $A(Ap^2)^2$

# The Fault-Tolerance Threshold

**Use *k* recursive levels of error correction**

Circuit failure

Gate failure

$$\frac{p_{fail}}{p_{th}} = \left(\frac{p_0}{p_{th}}\right)^{2^k}$$

Threshold

- **Error reduction is exponential in resources!**

# Assumptions Behind Fault-Tolerance

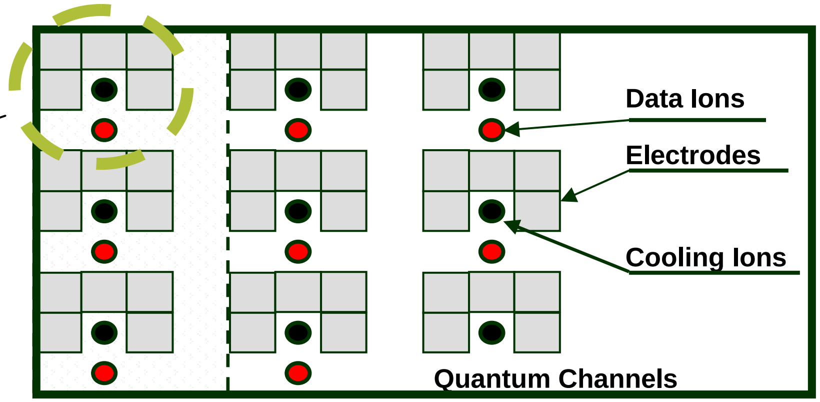
- **Classical logic must be faster than quantum**
- **Maximal parallelism is assumed**
- **Recursive quantum error correction**
- **Measurements in circuits**
- **Zero entropy source of qubits**
- **…**

# Outline

- How Does Quantum Computing Work?
- Large-Scale Architecture Requirements
- **A Quantum Logic Array Architecture (QLA)**
- Specialized Quantum Architecture (CQLA)
- Programming the Architecture
- Related work and other Architecture Models

# Physical QLA Structure (reconfigurable microarchitecture)

Basic Building Block



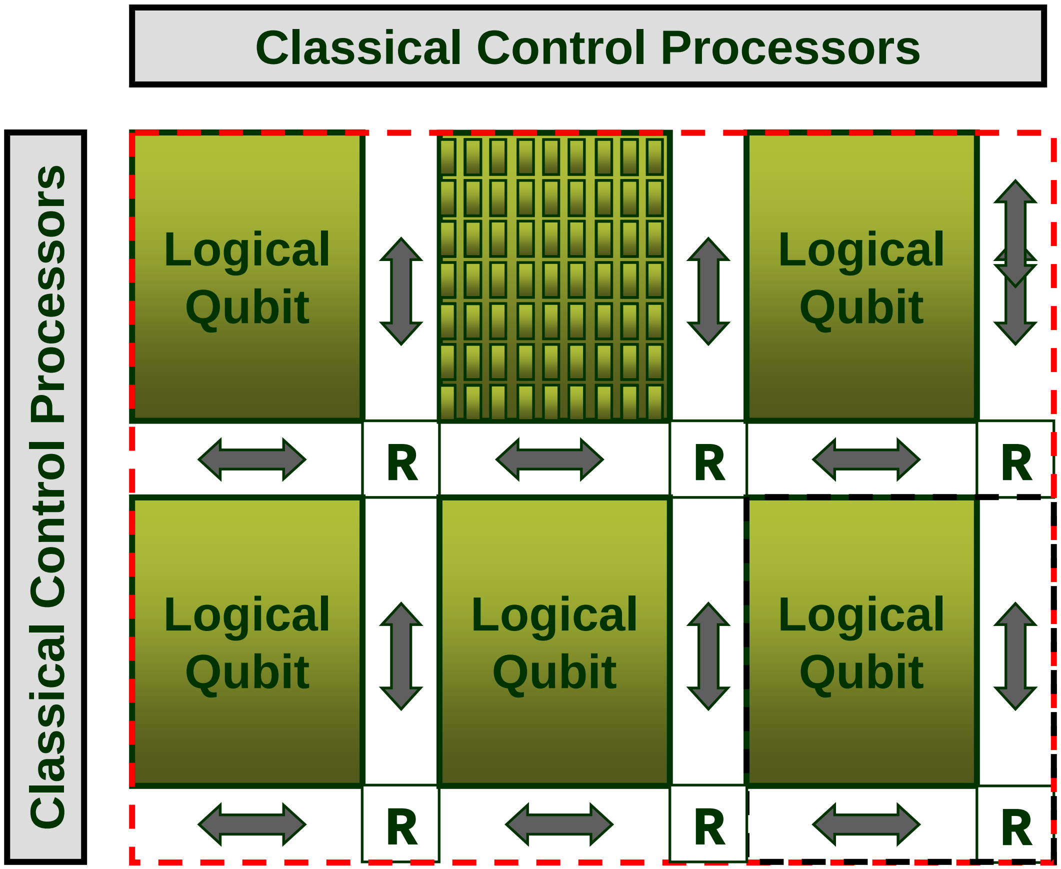

- Provides reconfigurability by allowing space for communication channels.

- Each building block consists of electrodes, the data ion, the sympathetic cooling ion, and free space around it to allow for the building of channels when the basic blocks are tiled together.

- The structure of a basic block is not fixed, but can be changed at design time to better accommodate the error correcting algorithms used. Computations are mapped onto the basic blocks at run-time.

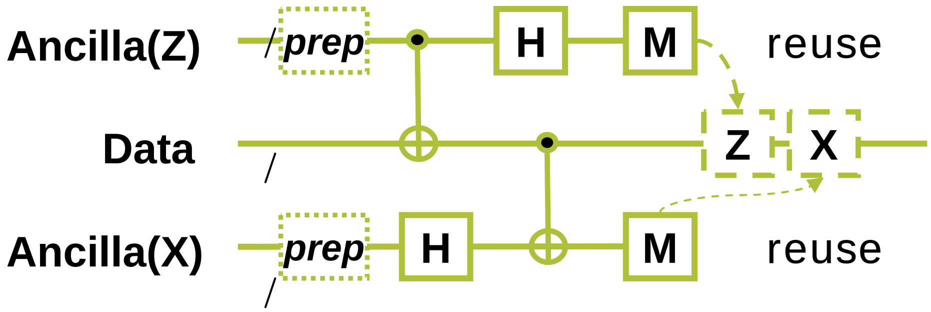
- Large-scale fault-tolerant architectures can be built by tiling basic blocks to form logical qubits and interconnect channels between them.

# The QLA: Tile-Based Computer

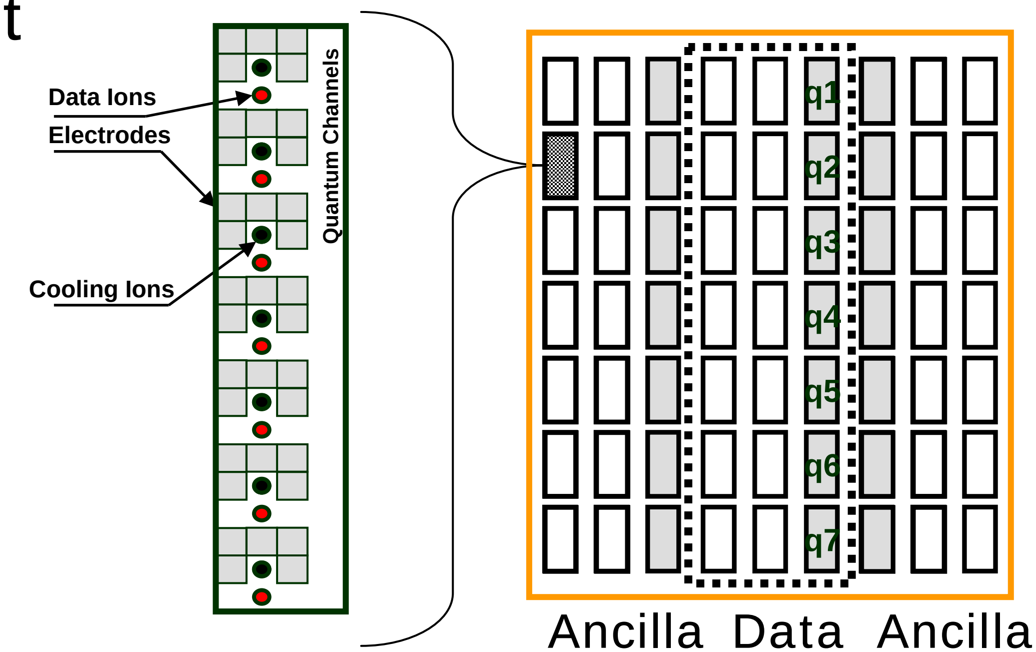


# The Logical Qubit (detail)

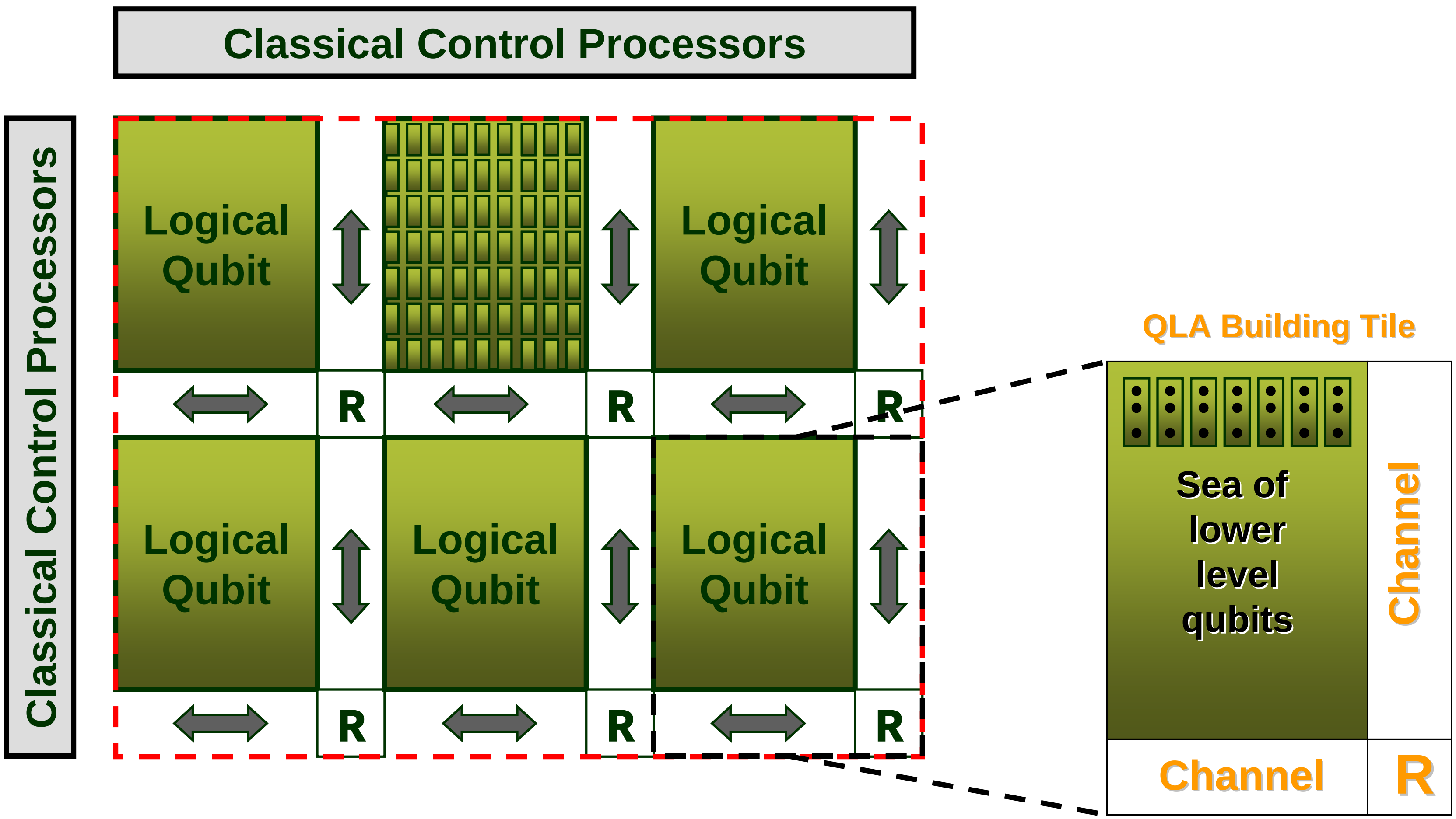
- Steane [7,1,3] error correction code



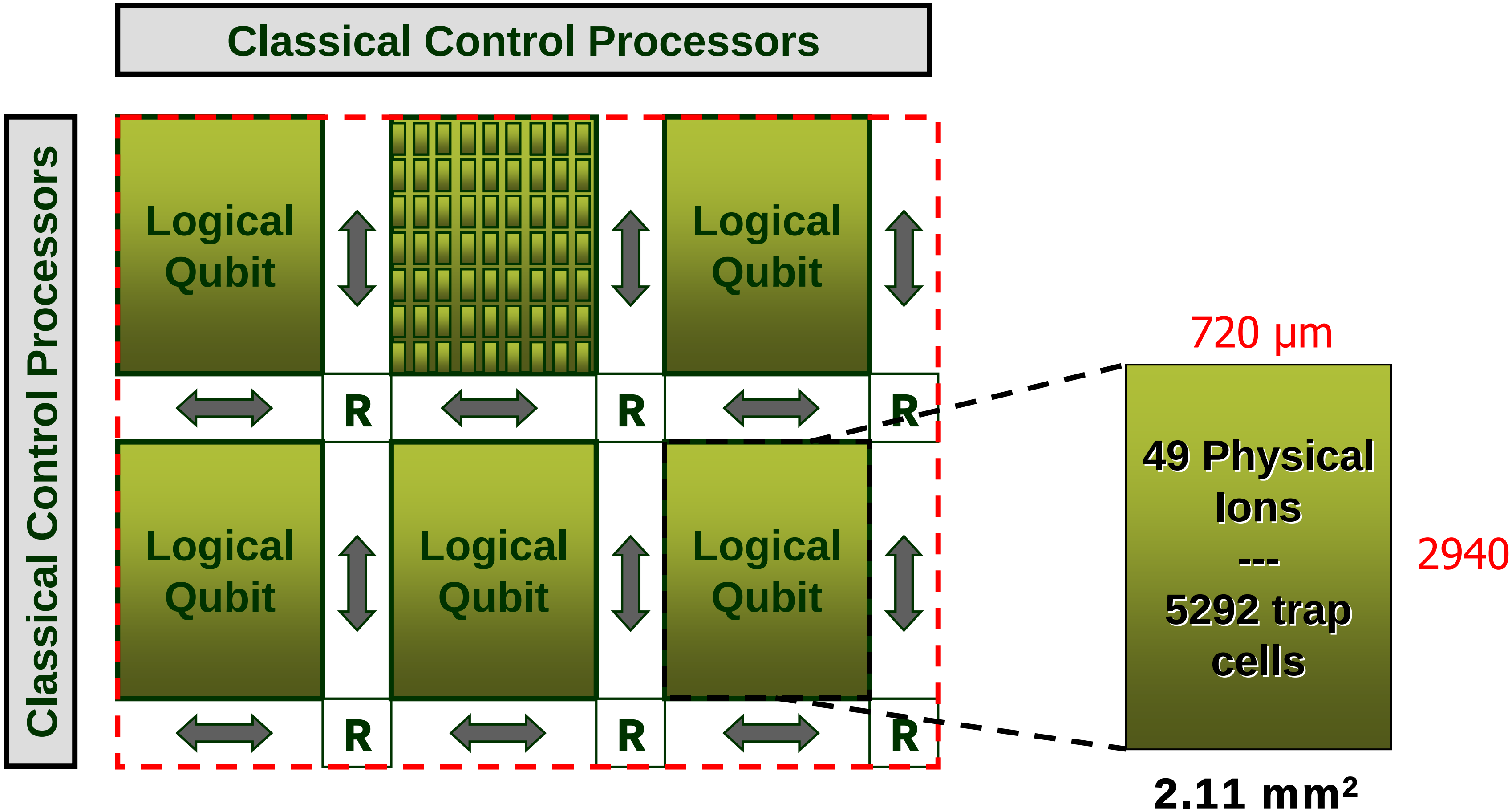

Level 2 Qubit Structure



# High Level Architecture Overview



# High Level Architecture Overview




~100 logical qubits per 90nm-technology Pentium 4 processor, compared to 55 million classical transistors within each such P4

# Question?

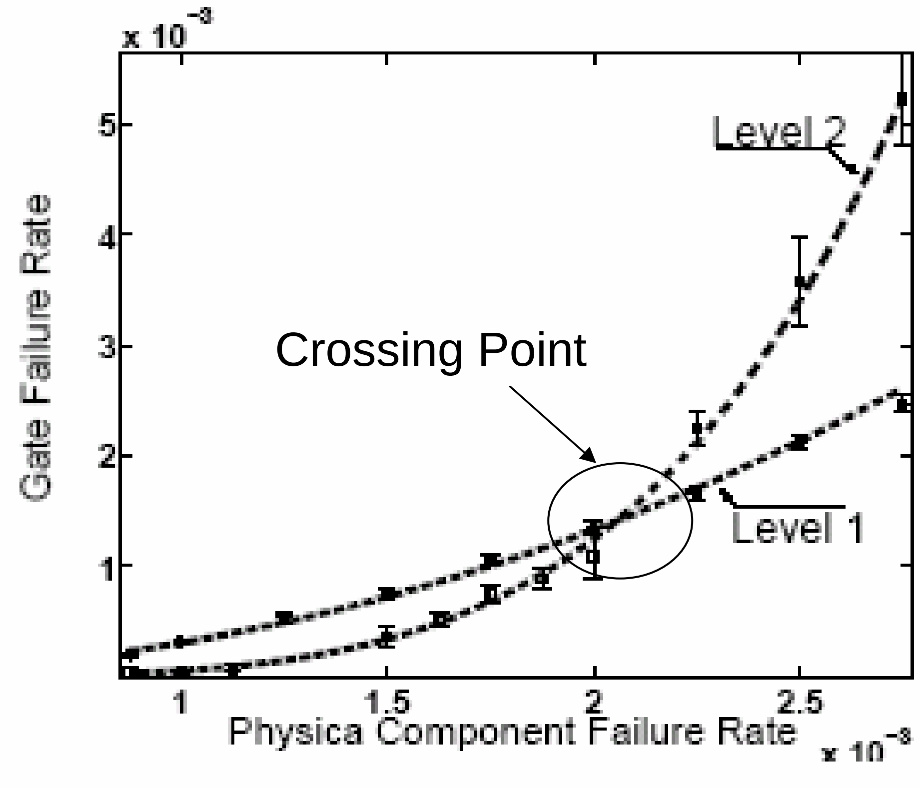
How feasible is the QLA? What is the real Fault-tolerance threshold for a *full* quantum information processing system ? (with all errors: gates, controls, wires…)

# Target:

- **Trapped ion quantum computer**

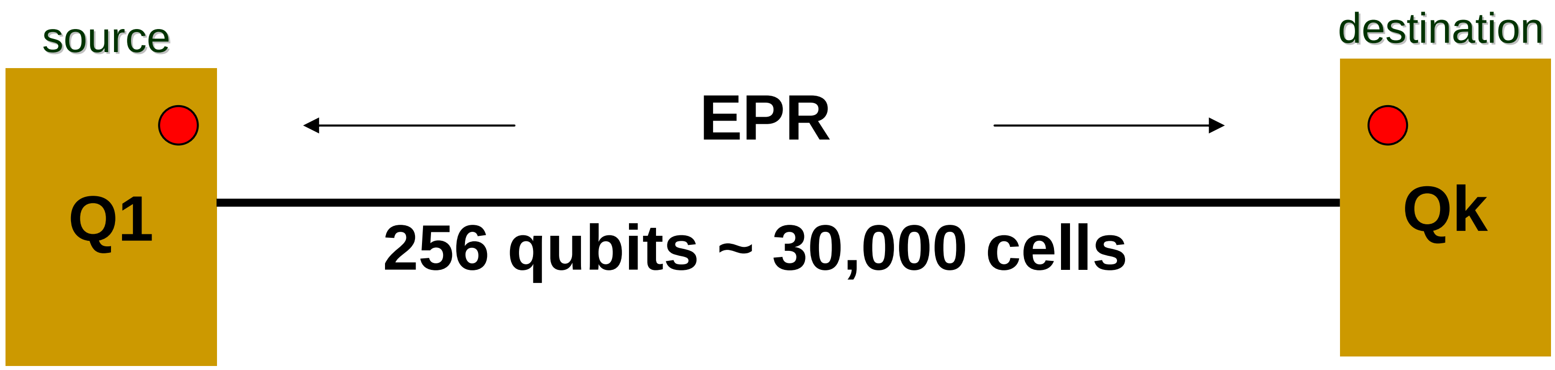
# Logical Qubit Threshold Results

- Ideal Ion-Trap Parameters: Crossing point was observed at $2.1 \times 10^{-3}$




- In real life: 4731 locations in a CNOT (11,188,815 pairs), ~60 hours simulation running time, 3,132,443 malignant pairs (~$3.2 \times 10^{-7}$)

# Quantum Resource Distribution




- Ballistic channels are too faulty for the data to move through at very large distances.

- We use the concept of teleportation developed by Bennet et. al. in 93, which employs entangled EPR pairs to recreate the state of an ion at the desired destination without physically moving the ion.

- The EPR pairs are purified upon arrival with the use of ancillary EPR pairs, which are constantly reinitialized to zero.

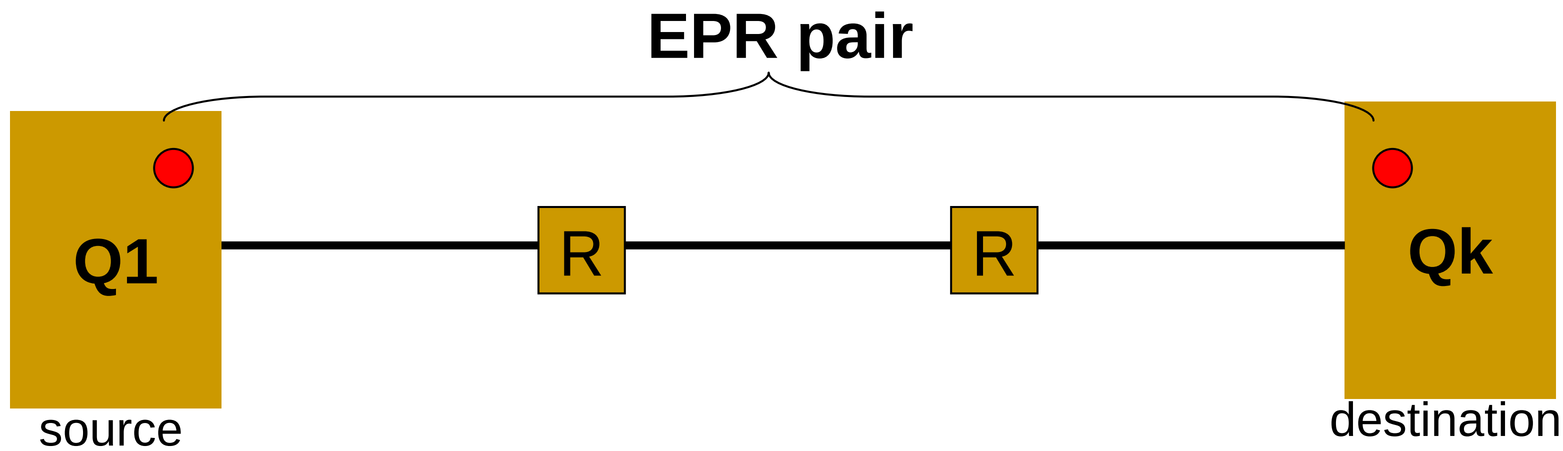
# Building a Quantum Architecture

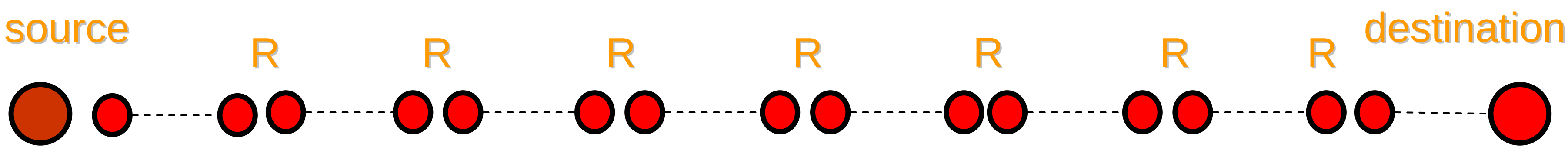
- Reliable and Realistic Technology
  - Reliable initialization of each qubit
  - Universal set of quantum operations
  - Ability to Measure the system

- Fault-Tolerant Layout and Error Correction

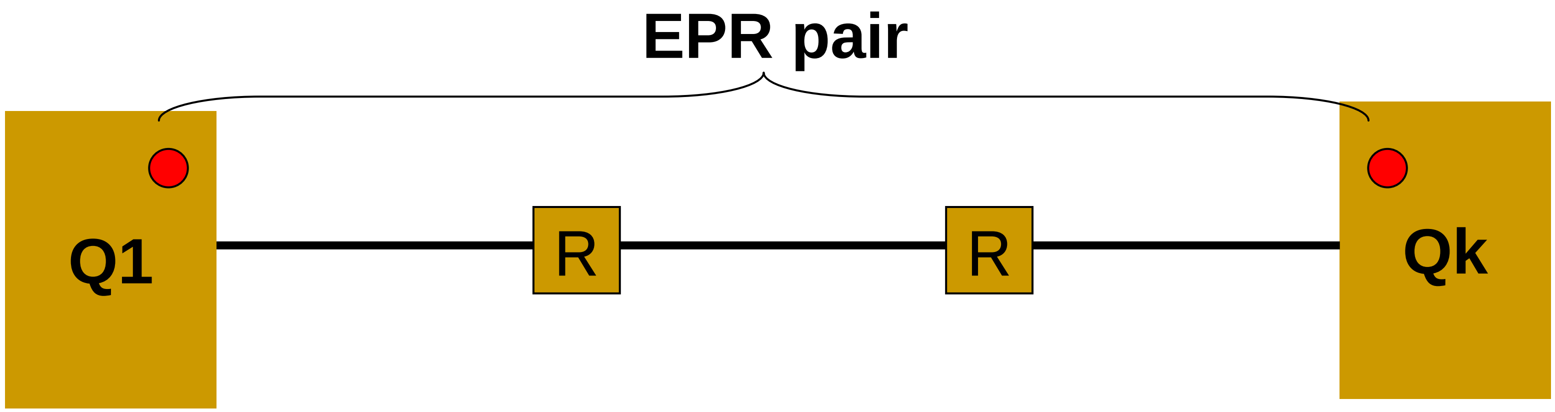
- **Efficient Quantum Resource Distributions.**

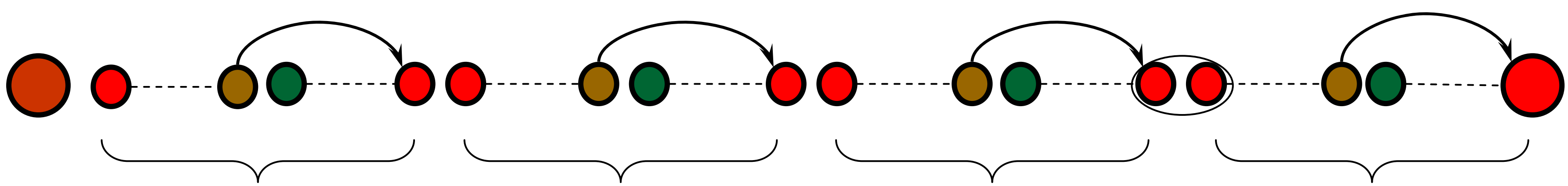
# Repeater stations solve the problem



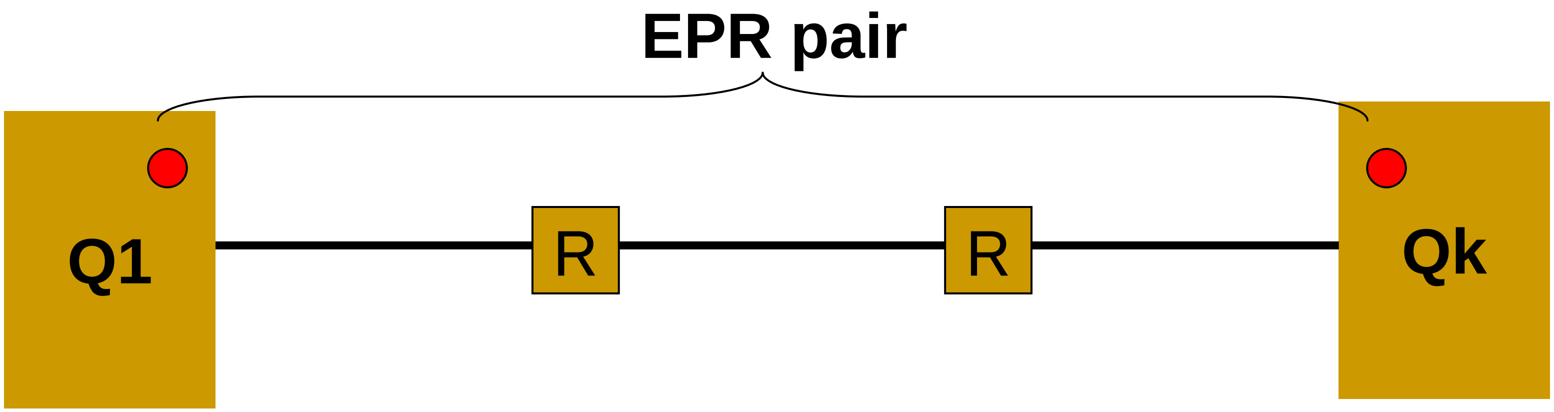



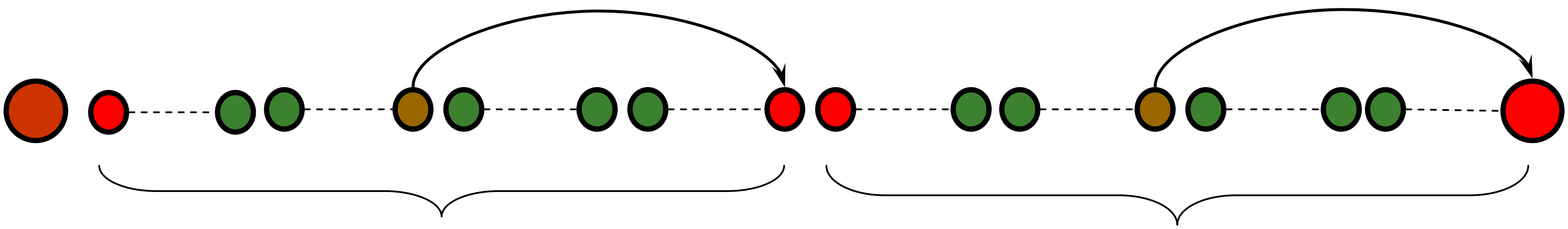
# Repeater stations solve the problem



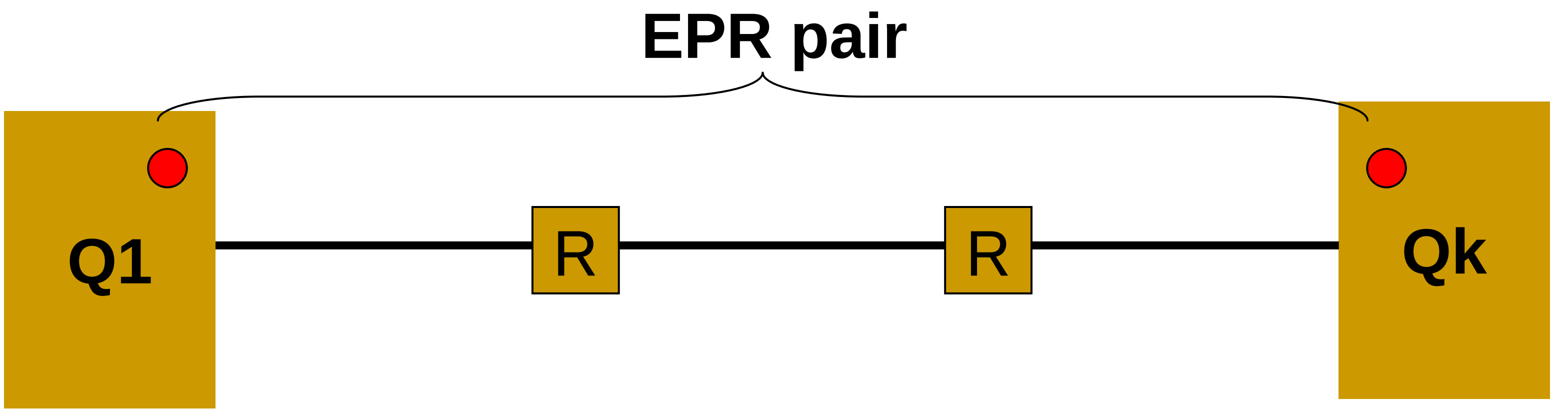



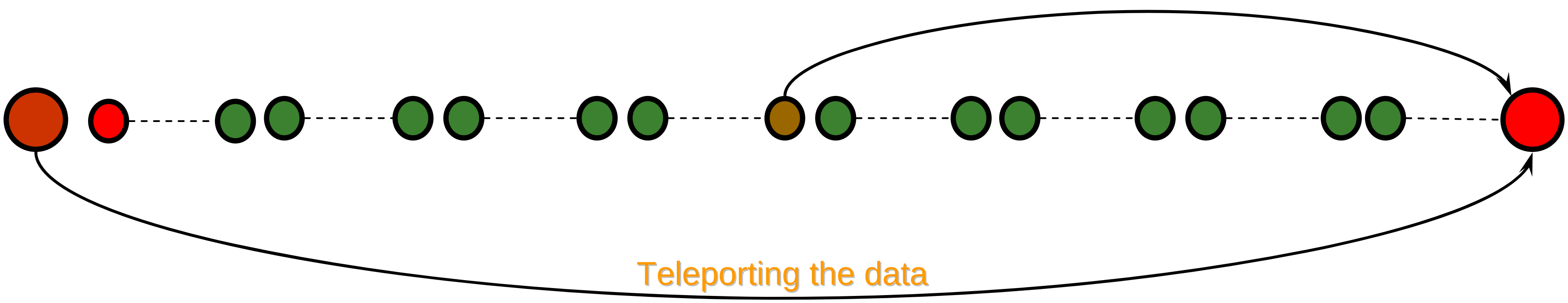
# Repeater stations solve the problem



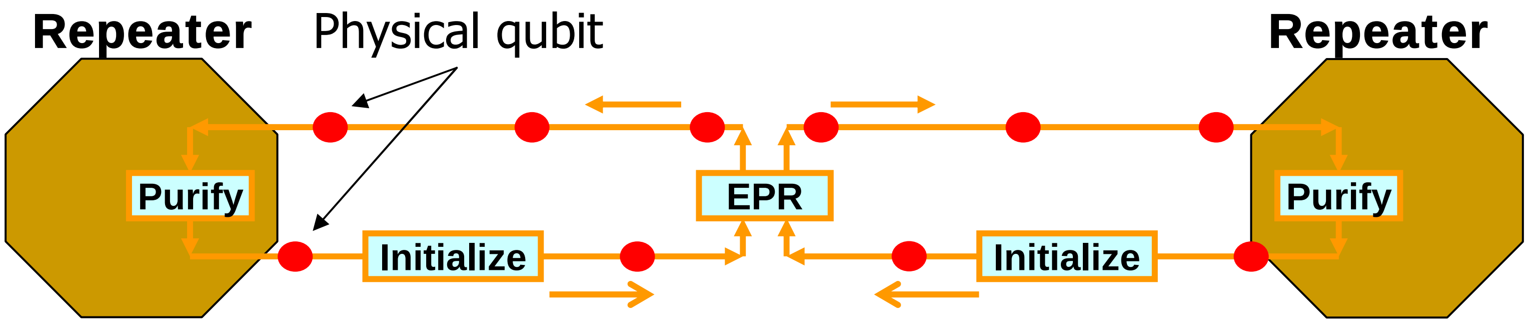



# Repeater stations solve the problem

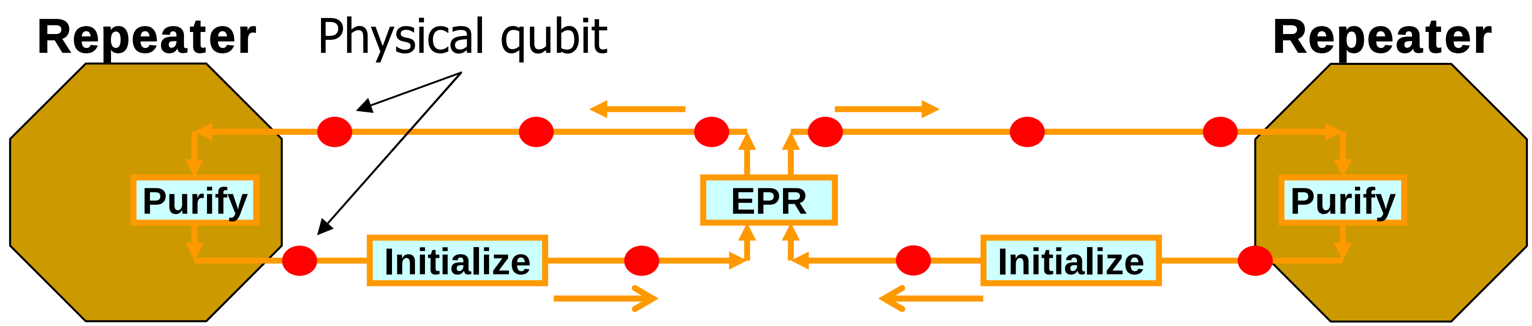


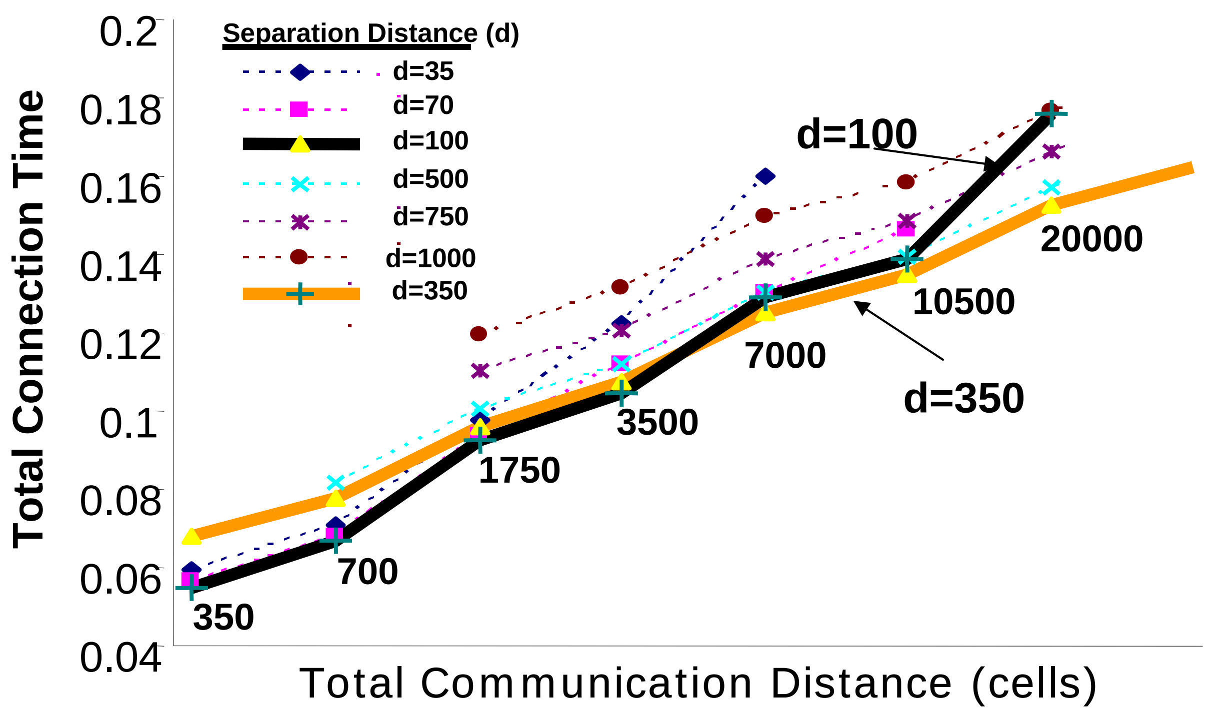



# Communication Channel: Detail



# Communication Channel: Detail

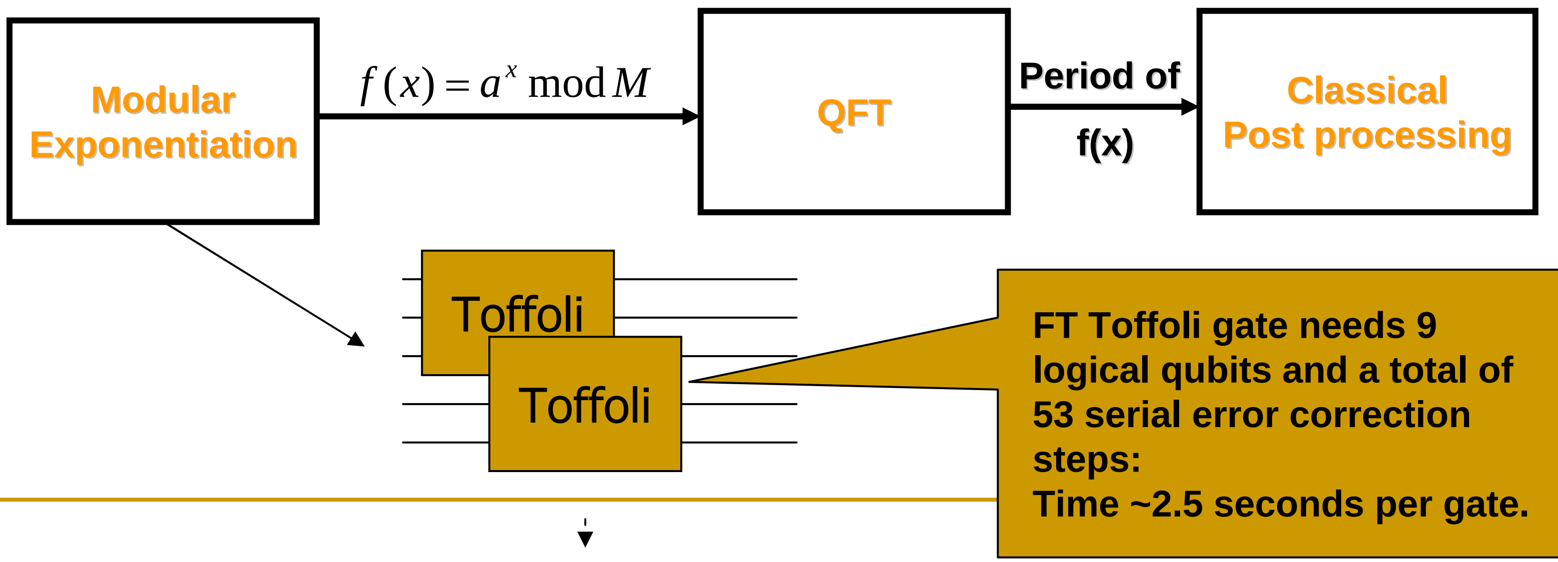


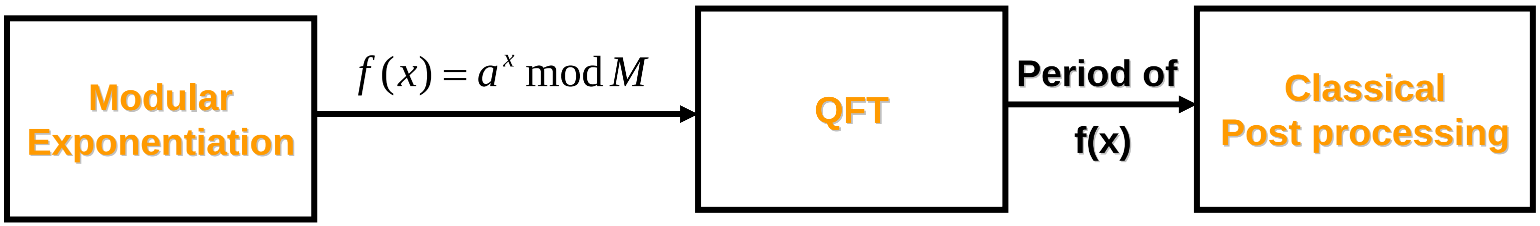
# Factoring and Integer

**Classical Factoring:** Exponential complexity. Cavallar in 2000 has demonstrated the factorization of a 512-bit number in seven calendar months on 300 fast workstations, two SGI Origin 2000 computers, and one Cray C916 Supercomputer - a process which amounts to 8400 MIPS years.

**Quantum Factoring:** Shor's Algorithm proposes polynomial time, however real time estimates currently don't exist due to the complexity of the system.



# Factoring and Integer (estimates)




- **128-bit:** 63,730 Toffoli Gates with 21 ECC steps per Toffoli for modular exponentiation. Thus we have 21(63,730)+QFT = 1.34 x $10^6$ time steps = ~ 16 hours. ➔ 16*1/.75 ➔ **~21 hours**
- **512-bit:** 397.910 Toffoli Gates + QFT ➔ **~5.5 days**
- **1024-bit:** 964,919 Toffoli Gates + QFT ➔ **~13.4 days**
- **2048-bit:** 2,301,767 Toffoli Gates + QFT ➔ **~32 days**

# Major QLA Problem!!!!!

**AREA and Classical Resource EXPLOSION**

# Major QLA Problem!!!!!

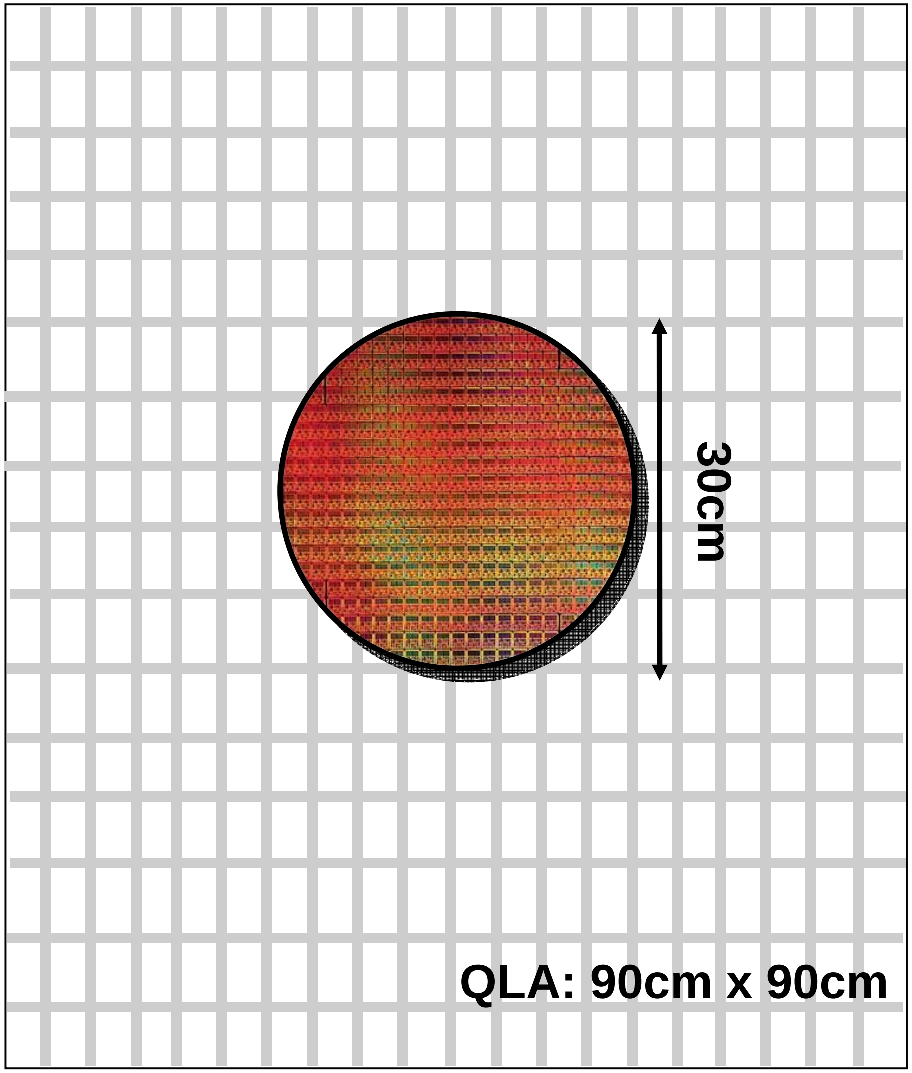
**AREA and**
**Classical Resource**
**EXPLOSION**

**Solution:** Specialized Architecture Elements?

# Outline

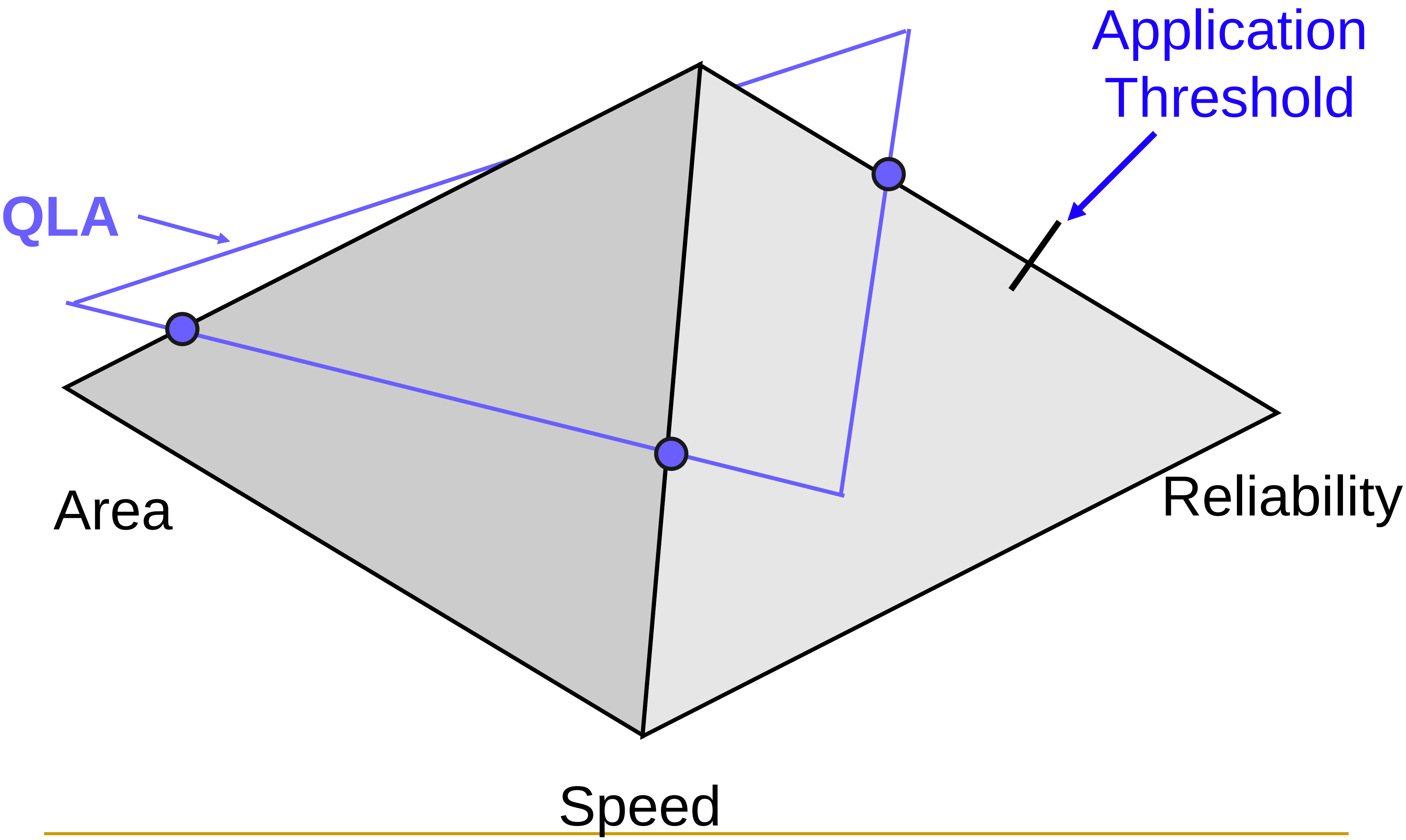
- How Does Quantum Computing Work?
- Large-Scale Architecture Requirements
- A Quantum Logic Array Architecture (QLA)
- **Specialized Quantum Architecture (CQLA)**
- Programming the Architecture
- Related work and other Architecture Models

# QLA - Revisited

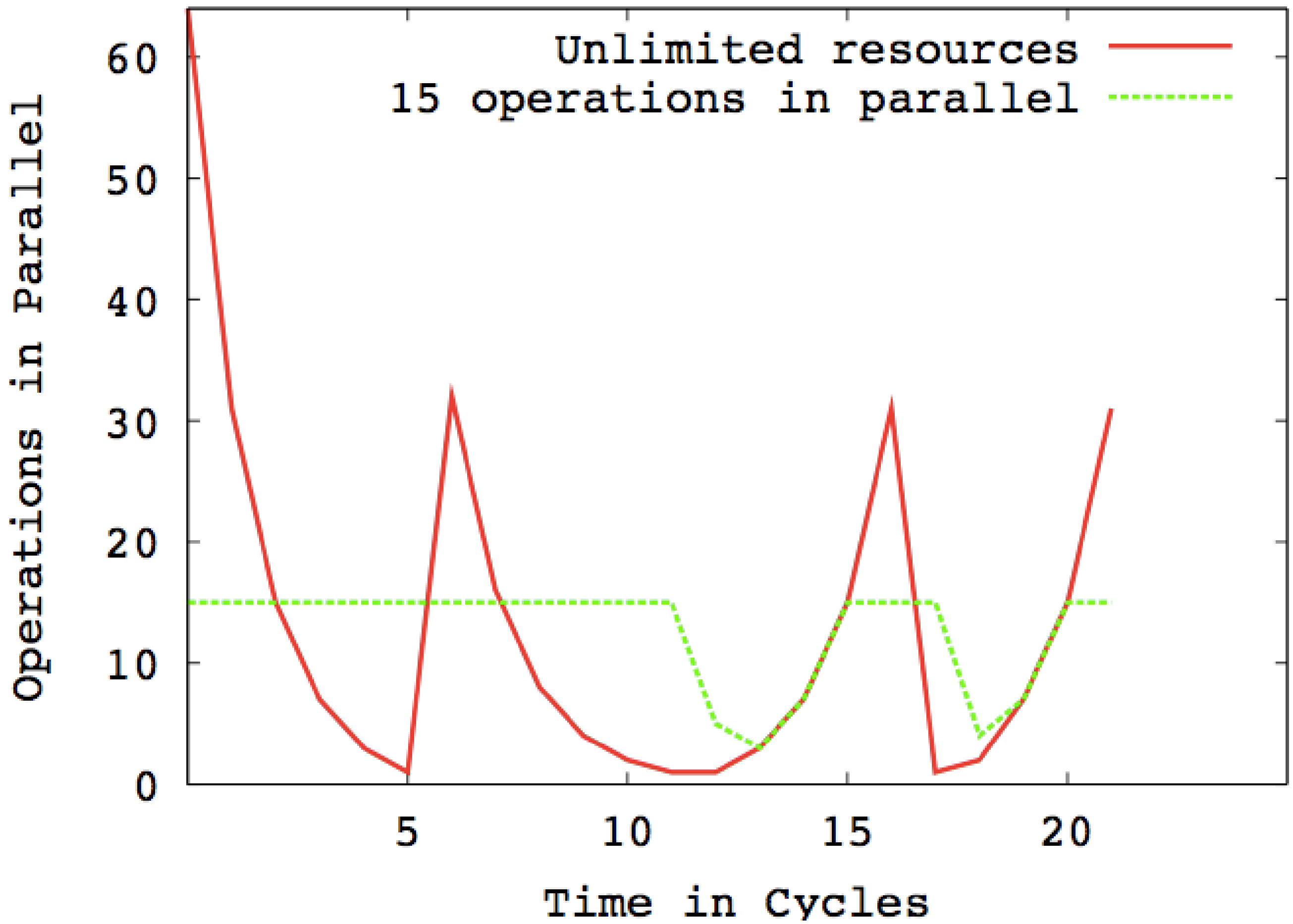



- **Conventional Wisdom:** Maximize parallelism necessary to minimize computation time and reduce probability of failure.
- **Compute anywhere:** All blocks logically equal

# Design Pyramid

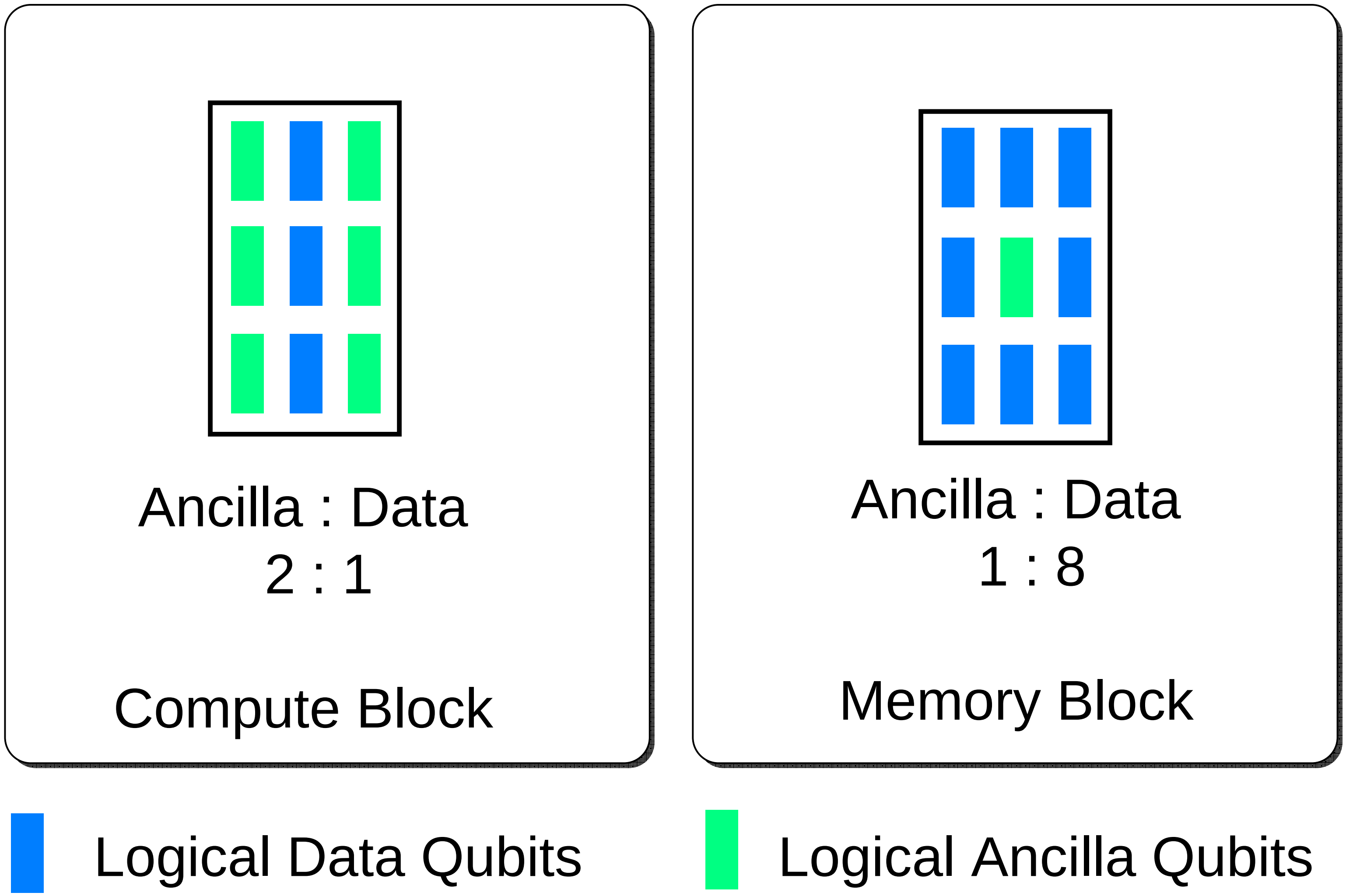


# Application Constrains Parallelism



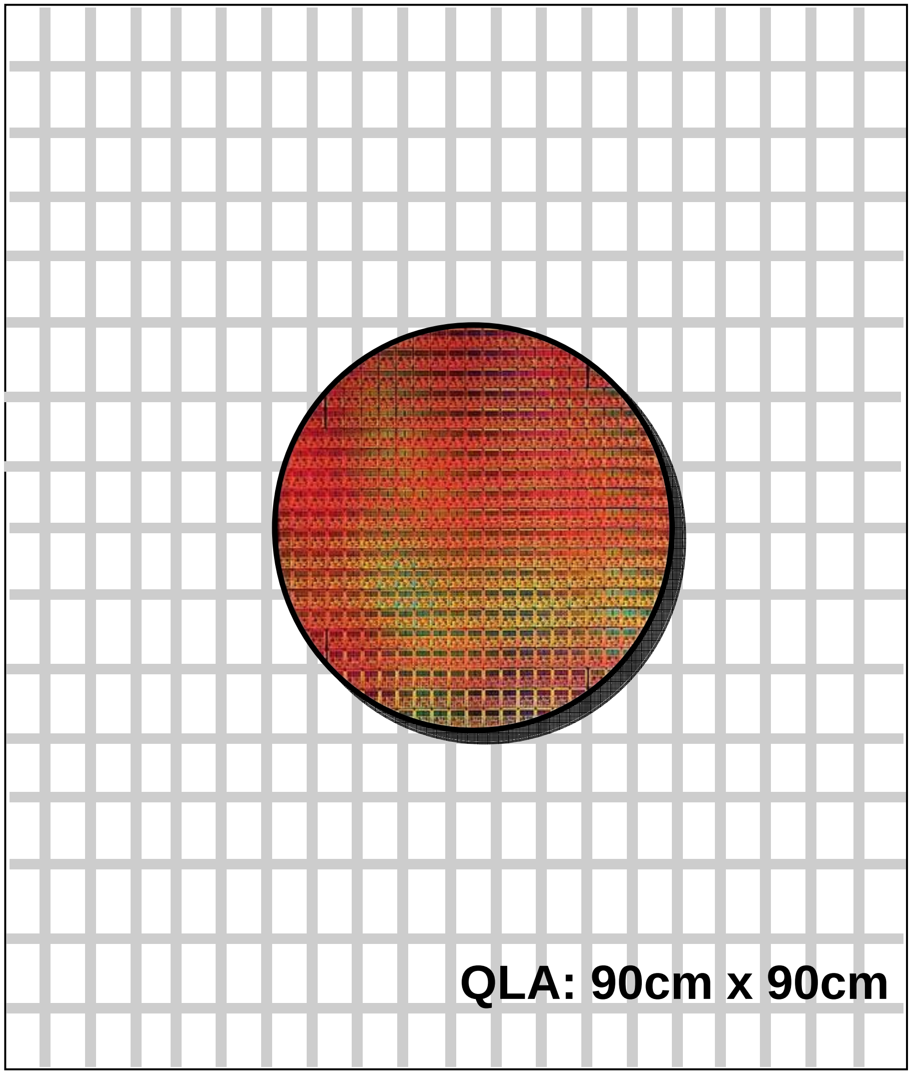

- <u>Modular Exponentiation Component</u>: **The Draper Carry-Lookahead Adder**

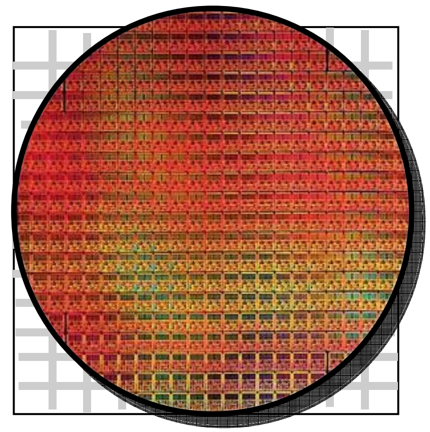
# Specialization



# Area Reduced

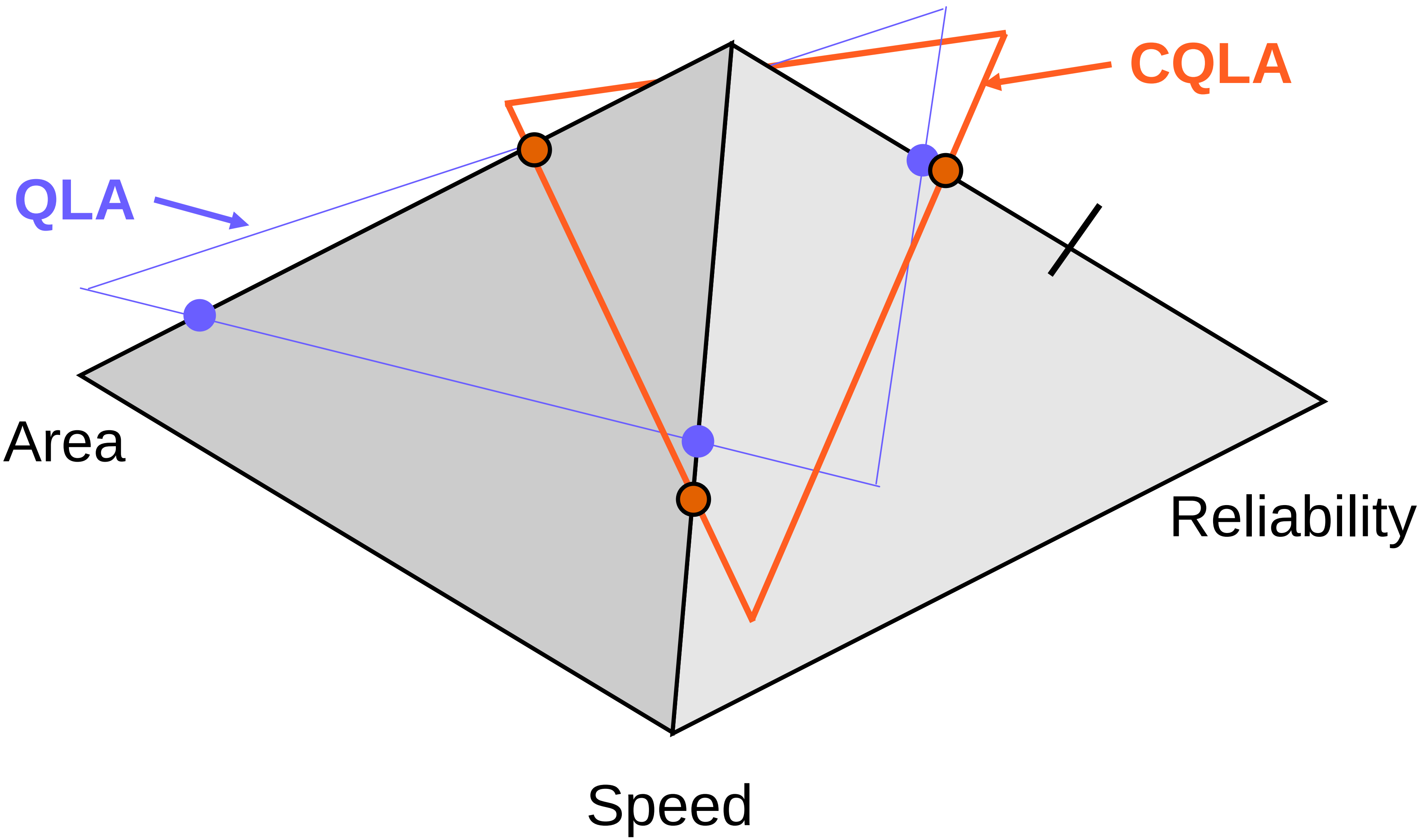
# Area Reduced



CQLA: 28cm x 28cm

# Design Pyramid - CQLA

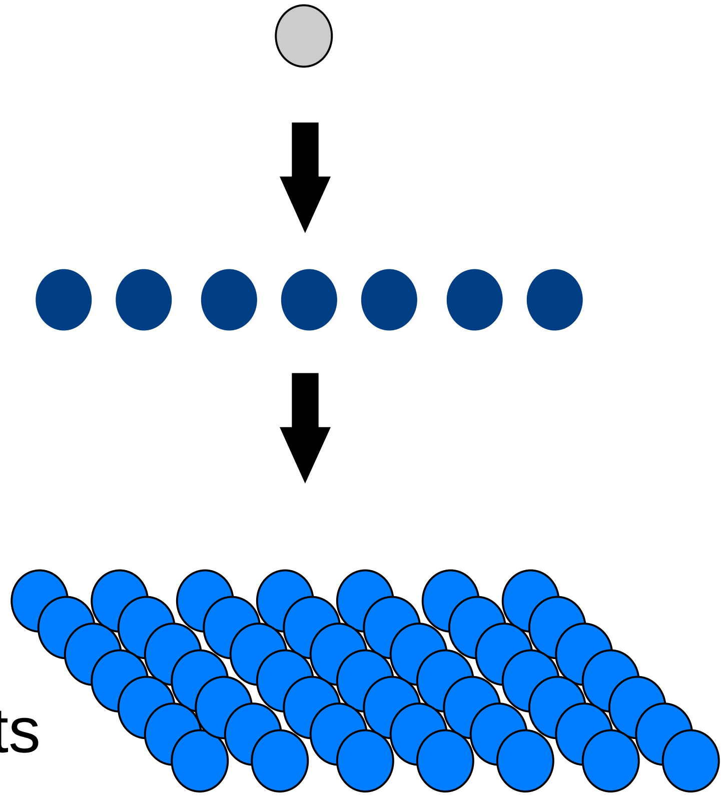


# Concatenated Codes

1 logical qubit

Level 1:
7 physical qubits

Level 2:
49 physical qubits



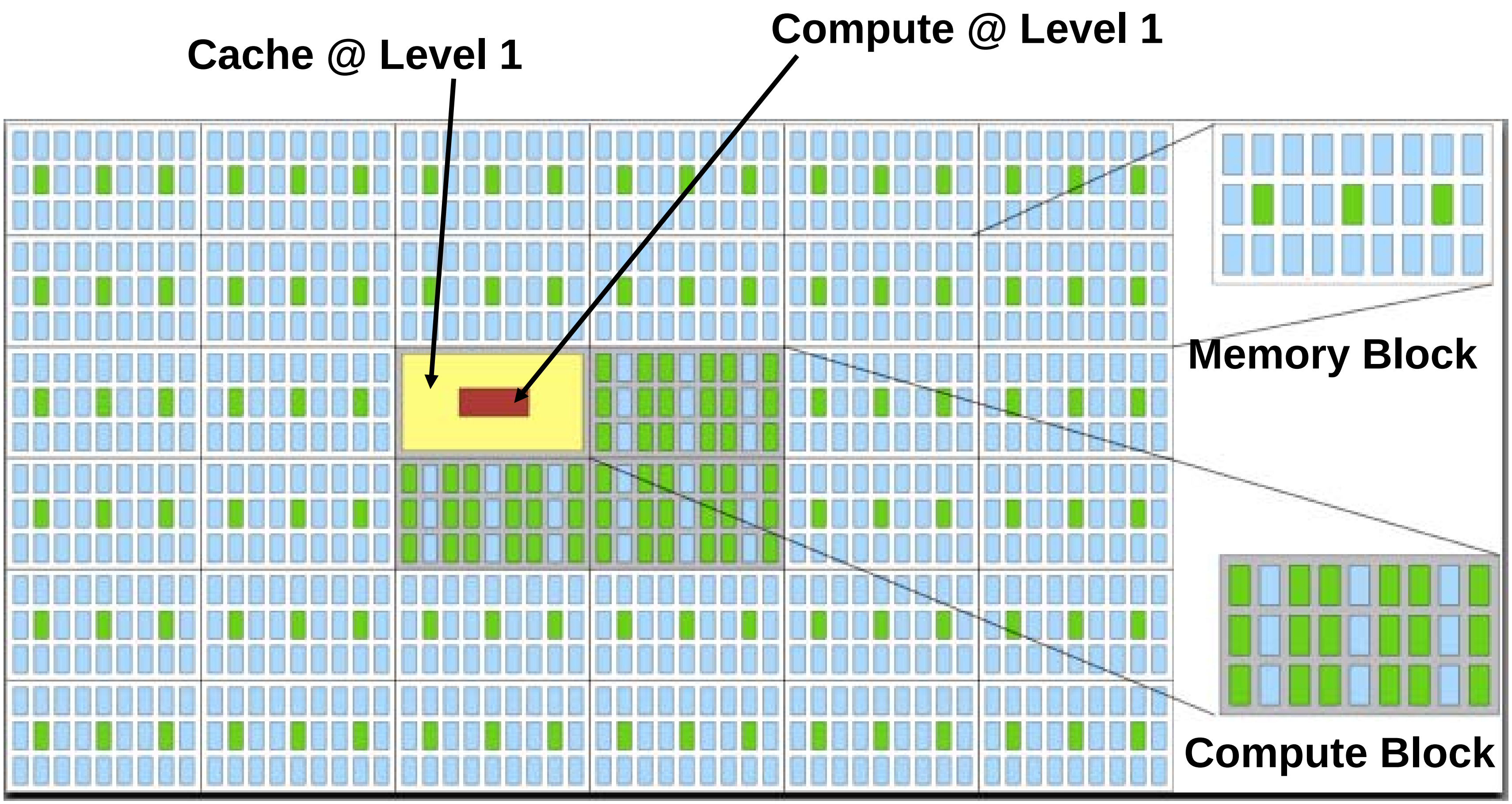
Reliability increases doubly exponentially.

Exponentially slower.

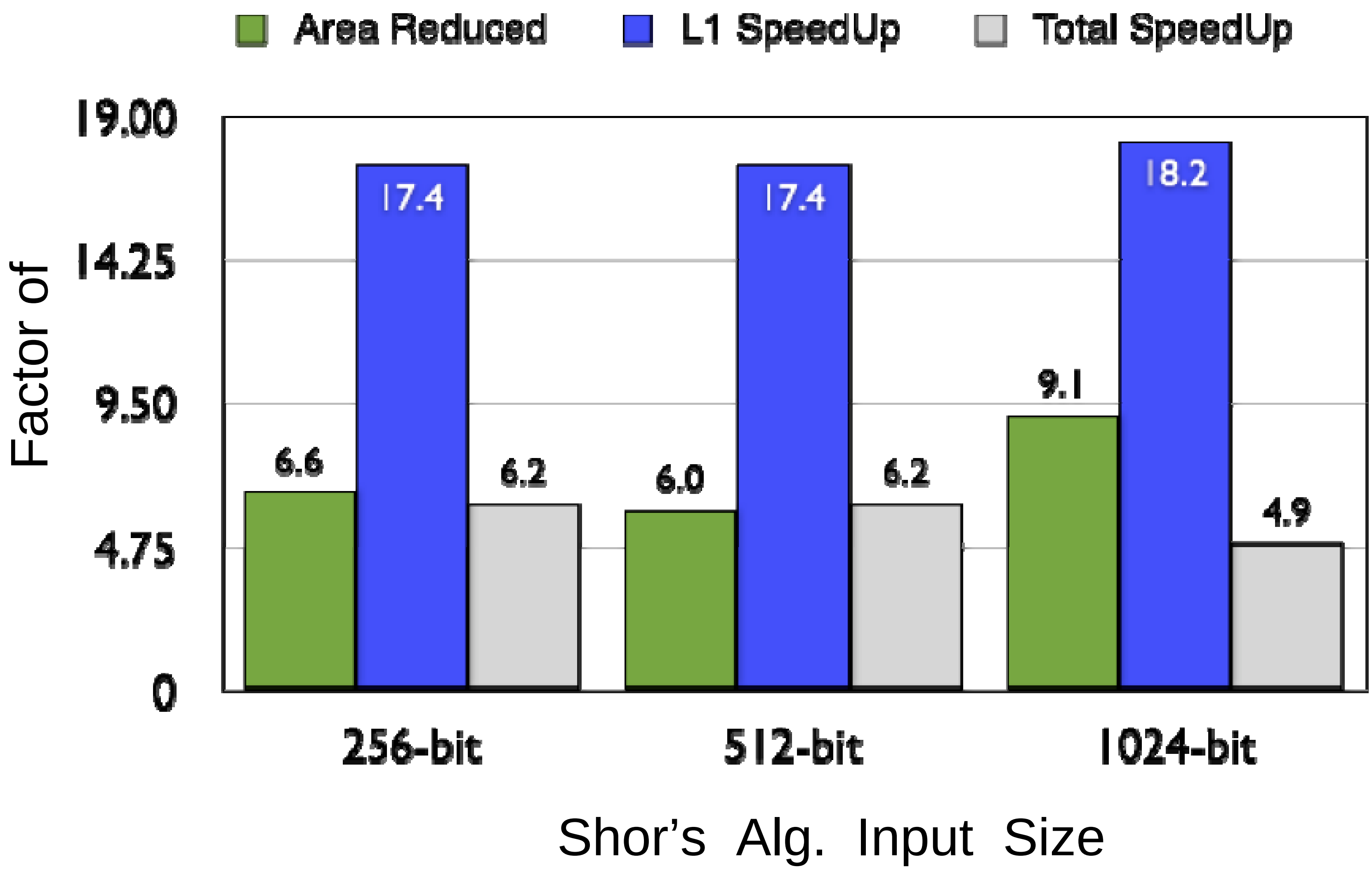
Exponentially greater resources.

Concatenated Steane Code

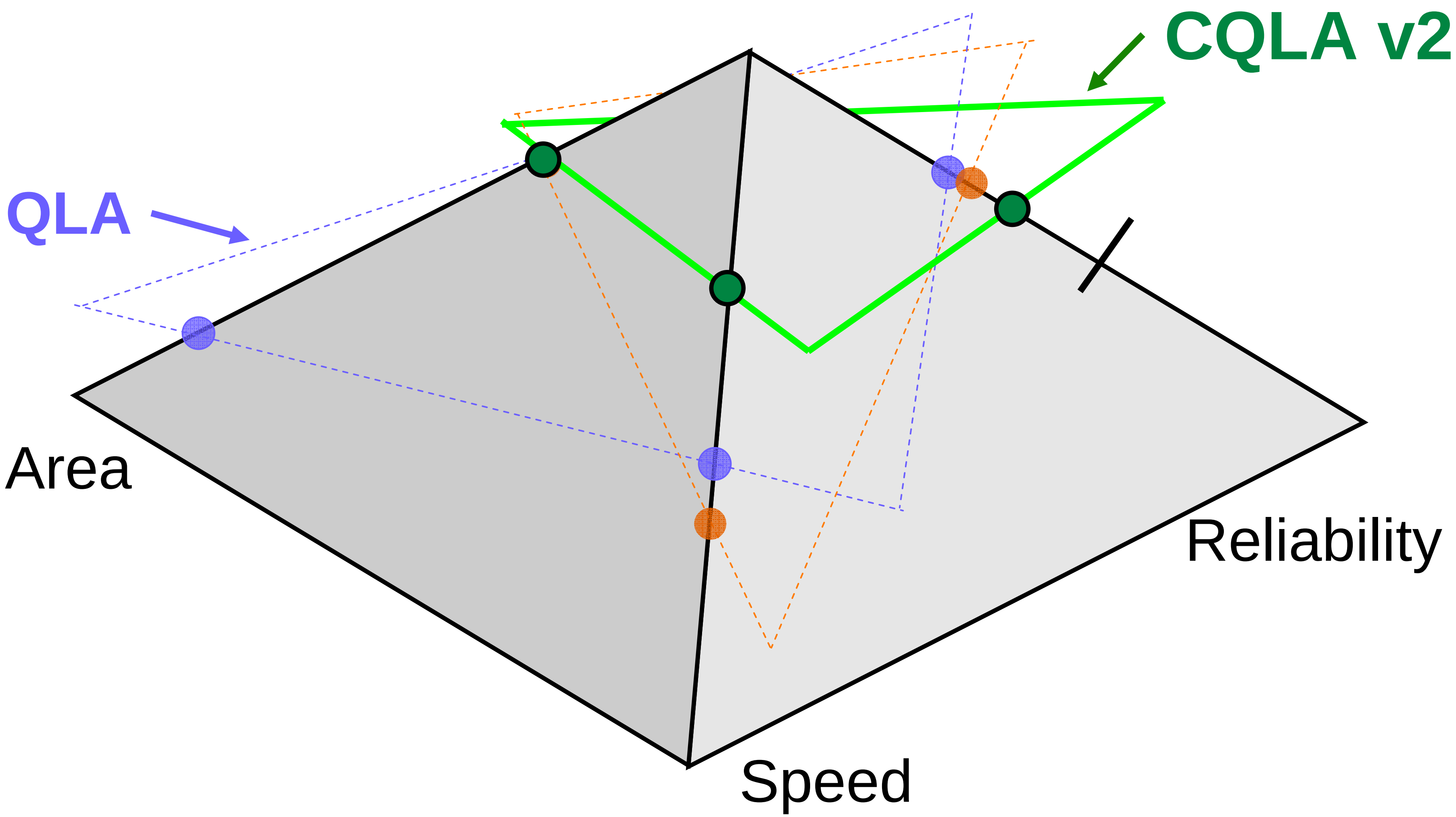
# Faster CQLA



# Overall Results



# Design Pyramid – CQLA v2



# Outline

- How Does Quantum Computing Work?
- Large-Scale Architecture Requirements
- A Quantum Logic Array Architecture (QLA)
- Specialized Quantum Architecture (CQLA)
- **Programming the Architecture**
- Related work and other Architecture Models

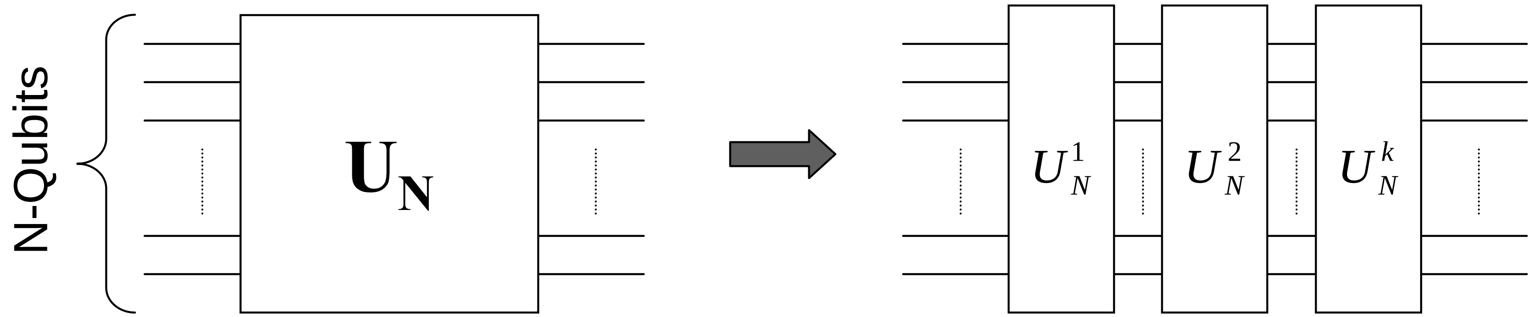
# Programming the Architecture

- A quantum compiler, much like a classical one, will require both technology dependent and technology independent optimization techniques.

- **GOAL:** Achieve the best possible schedule of a given class of gates for a given technology taking into account factors such as system microarchitecture, noise, gate execution time, and so on ...

# Programming the Architecture (Cont.)

- Logic Circuit Synthesis:



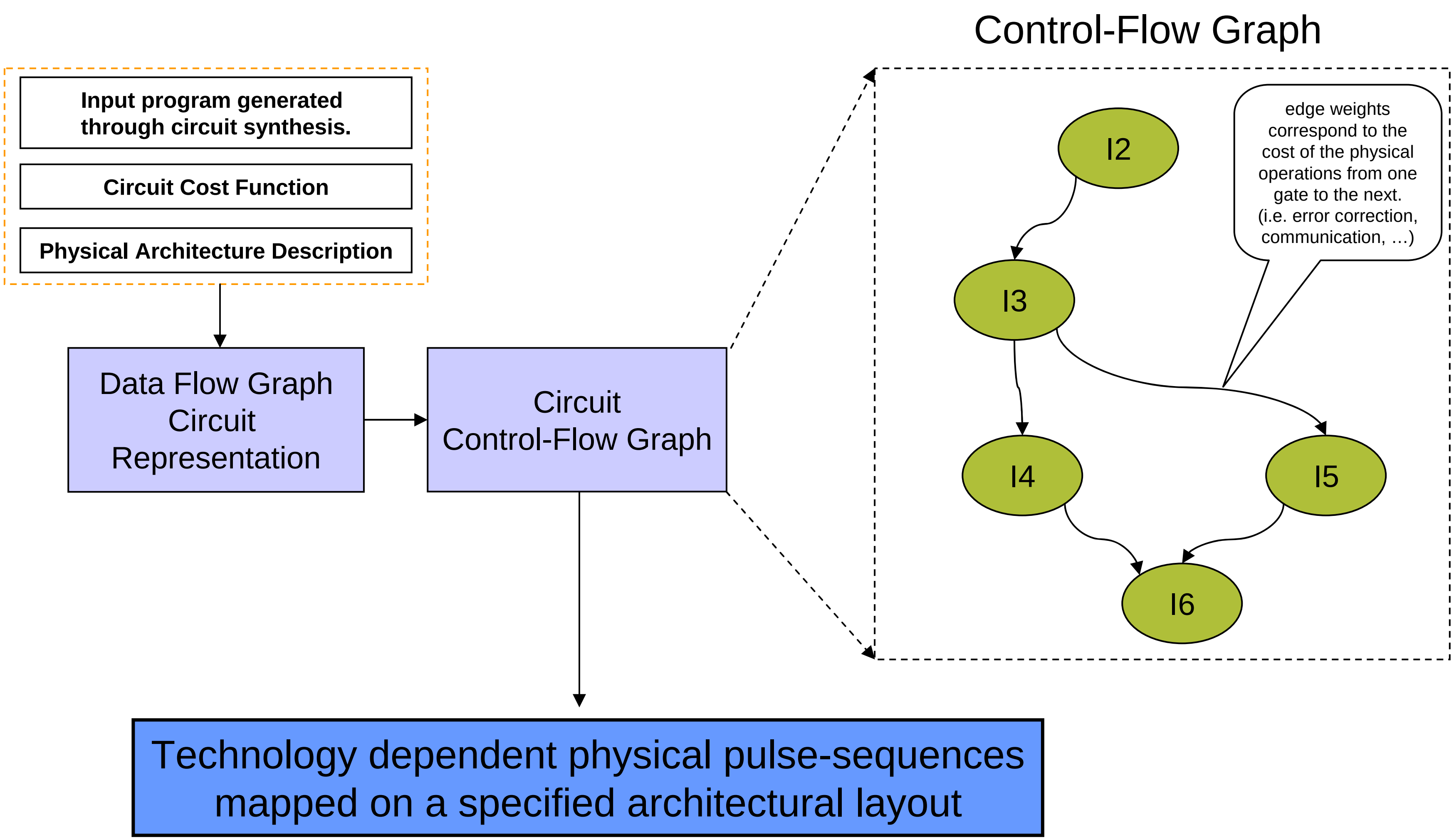

- Current QC technologies cannot implement gates larger than 1 and 2-qubits. All 1-qubit gates are easy with the most common 2-qubit gate: the CNOT gate.
- Lower bound of CNOT count for general unitary operator U is: $\frac{1}{4}\left(4^n - 3n - 1\right)$

# Programming the Architecture (Cont.)

- Once the sequence of operations is known, the QLA allows the full hardware resource orchestration for each individual application.
- Instruction scheduling becomes similar to instruction scheduling in current data-driven, tile-based machines such as the RAW microprocessor for example, where unlike standard microprocessors, the problem is to:

**Schedule both instruction and communication events: temporally and spatially.**

# Possible Compiler Overview



# Physical Instruction Scheduling

## Classical

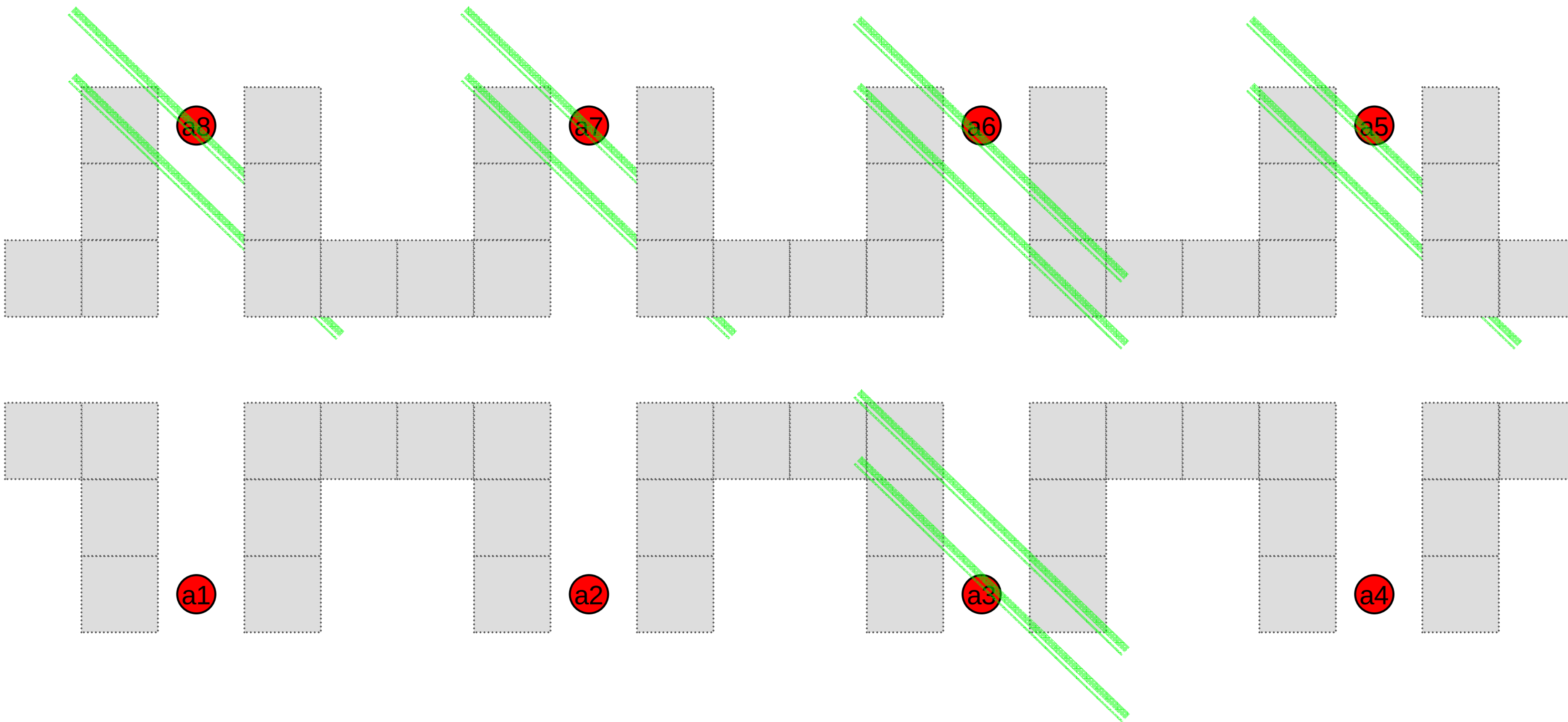
- How to best utilize the available physical resources given a set of instructions and computational units.
- Instructions are modeled using a control flow graph, where each node is an instruction and each edge indicates dependency.
- Find schedule for instructions to minimize some function (latency, area, power, …)
- Dependencies are based on operand location (i.e. RAW, WAR …)
- Physical layout is not an issue and operand communication is assumed instant. Data location does not affect instruction schedule.
- Highly advanced priority based heuristics exist including optimal superblock instruction scheduling that schedules 99% of the applications in < 1 second.

## Quantum

- How to best utilize the available physical resources given a set of instructions and computational units.
- Instructions are modeled using a control flow graph, where each node is an instruction and each edge indicates dependency.
- Find schedule for instructions to minimize some function (latency, area, power, …)
- Instruction dependencies are determined by commuting operation matrices, with the exception of measurement.
- Operand communication is slow, faulty, and highly dependent on the technology and the physical layout. Instruction schedule is *extremely* data driven.
- No existing scheduling algorithms exist, technology dependent or not.

# Physical Instruction Scheduling (Trapped Ions Example)

cnot a1,a8
cnot a6,a2
cnot a7,a2
cnot a3,a6
cnot a3,a5
cnot a4,a5
cnot a2,a1



# A Physical Scheduling Heuristic

- **Control-Flow Graph Generation and Instruction Priority Calculation.**
- **Disambiguation between source qubit and destination qubit for each available 2-qubit gate.**
- **Path generation and execution path choice relying on a) instruction priority; b) path interference with other chosen paths.**
- **Elimination of introduced stall cycles during deadlock detection and avoidance.**
- **Output a physical quantum assembly code whose execution time, failure rate, and circuit layout can be used as part of the edge-waits in the logical instructions control-flow graph.**

# Outline

- How Does Quantum Computing Work?
- Large-Scale Architecture Requirements
- A Quantum Logic Array Architecture (QLA)
- Specialized Quantum Architecture (CQLA)
- Programming the Architecture
- **Related work and other Architecture Models**

# Current Quantum Architecture Work

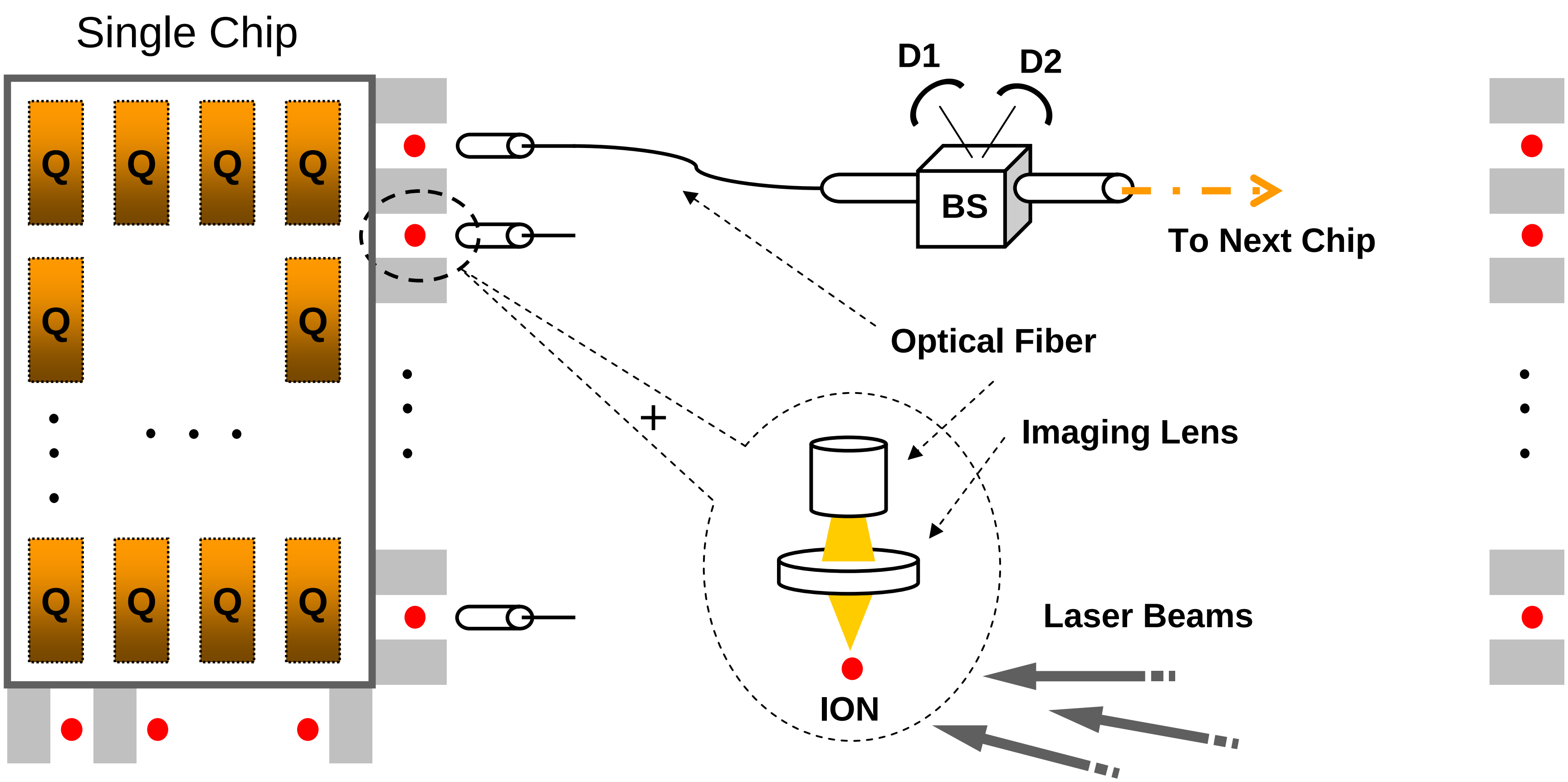
- E. Knill: leads the effort in teleportation-based error correction techniques and logical gate implementation.
- M. Oskin and D. Bacon study and model quantum architectures based on the most efficient error correcting codes known. Codes that utilize as much fault-tolerant network construction as the underlying physics of the device.
- M. Vladutiu, Romania uses reconfigurable circuit structures to improve quantum error correction scalability.
- Teleportation based distributed quantum systems for large-scale quantum applications are being studied by R. Van Meter at Keio University, Japan.

# Current Quantum Architecture Work

- I. Markov (U. of Michigan), A. Aho (Columbia), and I. Chuang (MIT) have studied in depth quantum circuit synthesis and the development of fault-tolerant software architecture for modeling quantum computers.
- T.N. Vijaykumar has proposed a method for further optimizing the QLA architecture through novel technique to parallelize quantum circuits.
- … In general, the field of QC is a fast growing field offering physicists, device engineers and computer scientists vast amount of resources for innovation.

# THE END

# Multi-Chip Area Example




To ion-trap chips are connected through an optical fiber network, where collected photons into a Beam Splitter (BS) station from two remote ions are measured forcing the ions into an entangled state. After the entanglement procedure we can teleport data ions from one chip to the next.