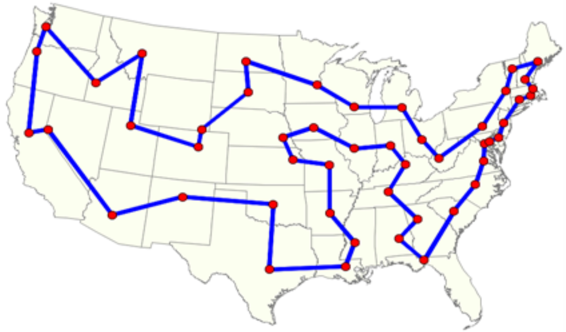
How about quantum computing?

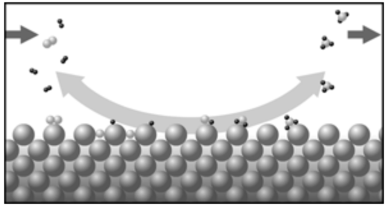
Bert de Jong
wadejong@lbl.gov

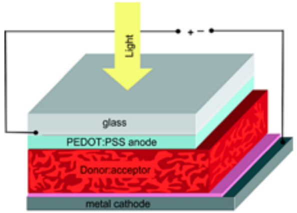
What makes quantum computing so exciting?

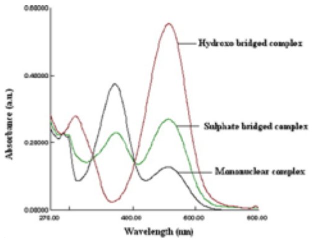
- Speedups over classical computing
- “Unbreakable” encryption protocols
- Quantum simulation
- Efficient optimization algorithms



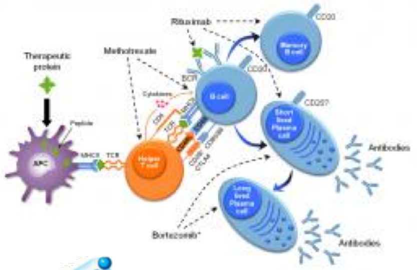
Why is a computational chemist like me interested in QC?

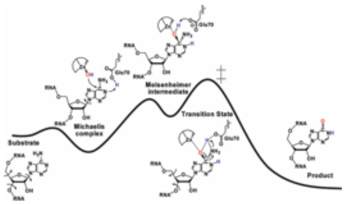






Understanding → Control

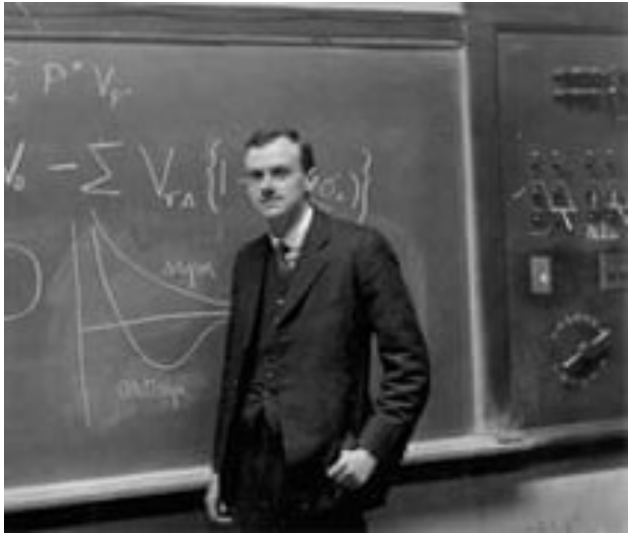




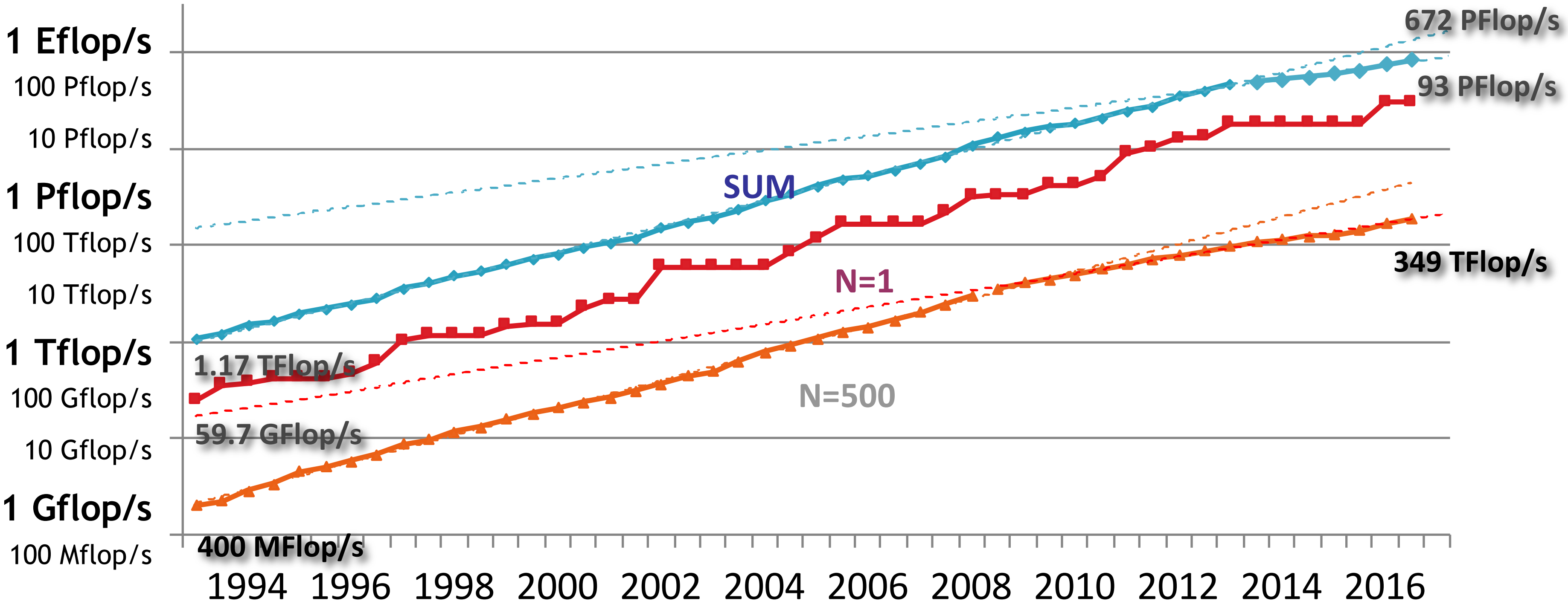
Challenge on classical computers is exponential complexity

The underlying physical laws necessary for the mathematical theory of a large part of physics and **the whole of chemistry** are thus completely known, and the difficulty is only that the exact application of these laws leads to ***equations much too complicated to be soluble***

Paul Dirac



Exaflop gives us only a factor of 10x … we need a lot more



Quantum chemistry on quantum computers

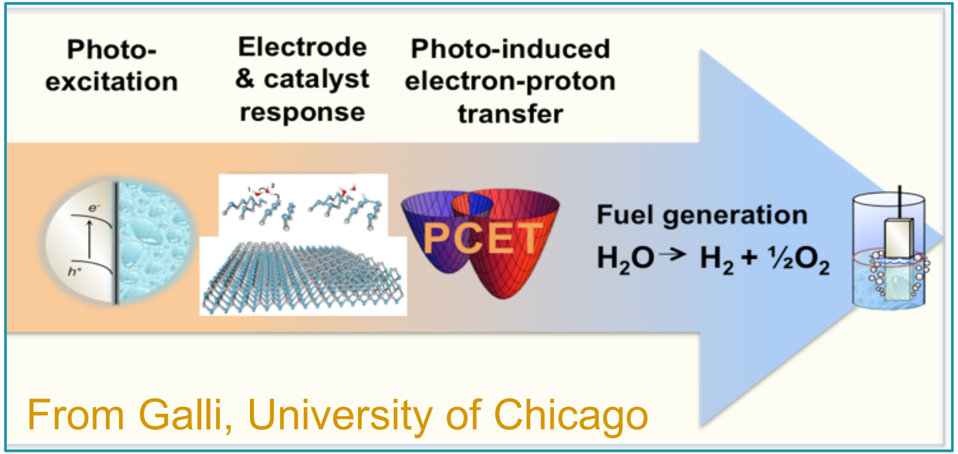
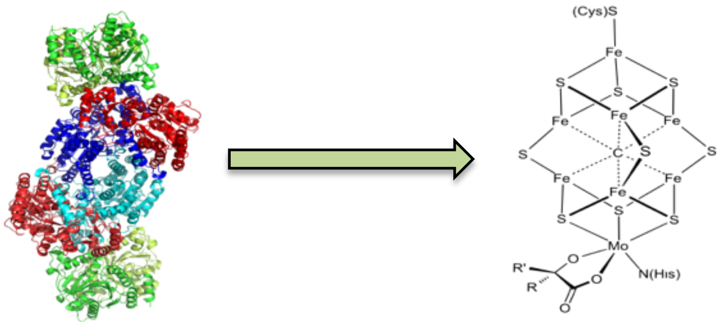


Photo-induced catalysis of water

Nitrogenase enzyme

$$N_2 + 3\ H_2 \rightarrow 2\ NH_3$$

"FeMoco"



Nature's answer to Haber Process

Inaccessible, even at exascale!
Quantum computer requires ~100 ideal qubits for solution

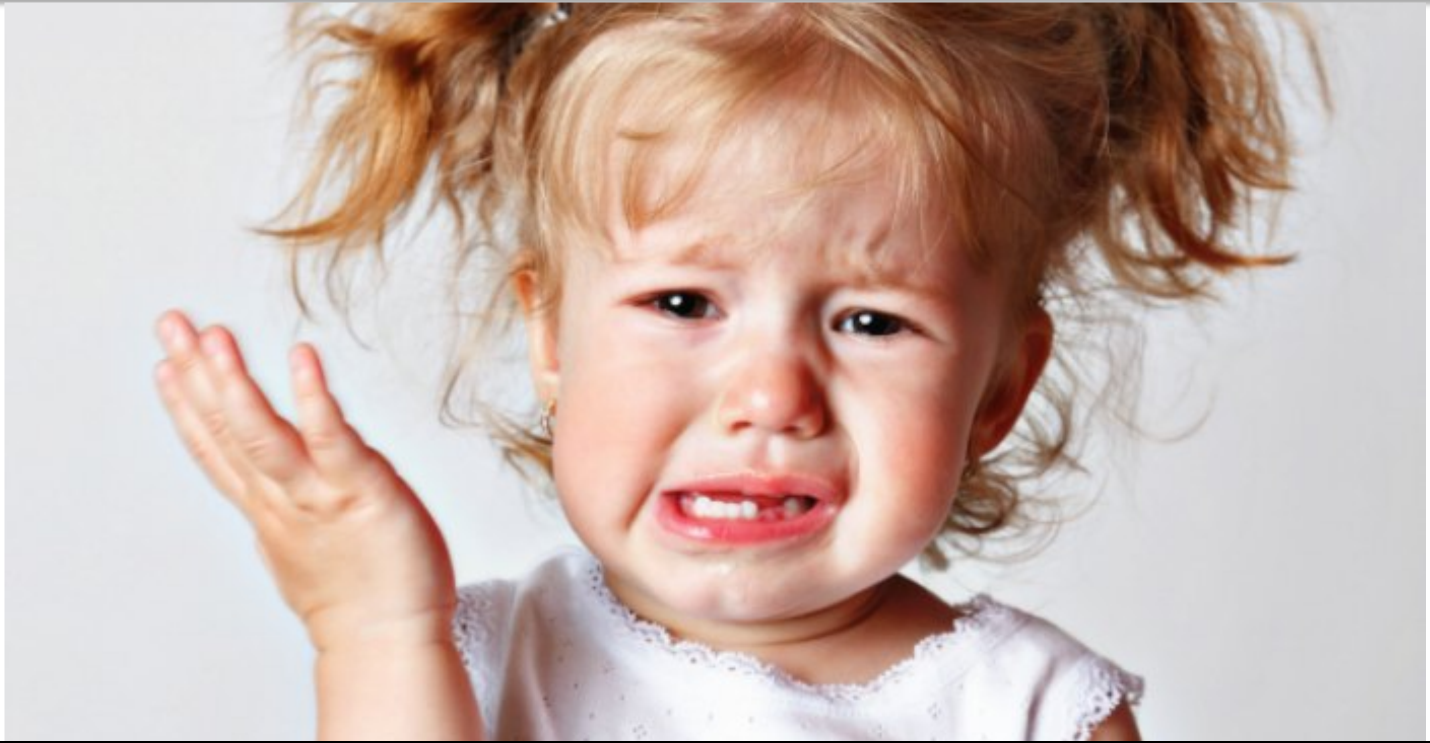
Behold the power of quantum computers

- 2^n complex coefficients describe the state of a composite quantum system with n qubits

- 100 qubits = 2^{100} states

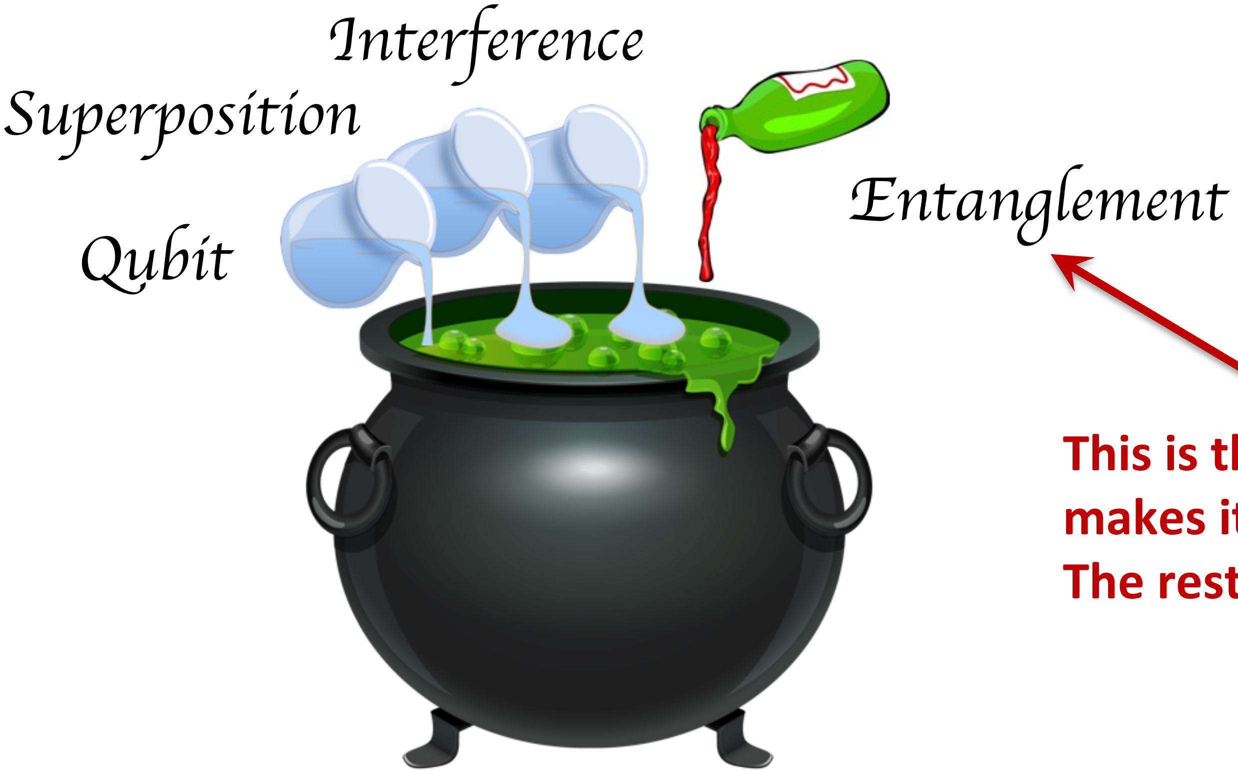
- Quickly reaches number of particles in the universe

How do you get into quantum computing?



You need to learn some physics (quantum mechanics) if you want to do quantum computing.

Ingredients to make a quantum computer work



This is the one that makes it "Quantum." The rest is just math.

You'll need to know some linear algebra…

Qubit state is represented as a two-dimensional *state space* in $\mathbb{C}^2$ with *orthonormal basis* vectors

State → wave function → $|\psi\rangle = a|0\rangle + b|1\rangle$ **→ *a* and *b* are complex**

$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ **and** $|1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ **are computational basis**

$|\psi\rangle = a|0\rangle + b|1\rangle = \begin{bmatrix} a \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ b \end{bmatrix} = \begin{bmatrix} a \\ b \end{bmatrix}$ **with** $|a|^2+|b|^2= 1$

Tensor products key for multiple qubits

- **Notation for two qubits**

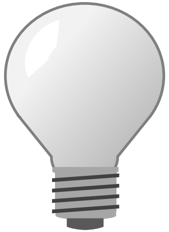
$$|0\rangle|0\rangle = |00\rangle$$

$$|\psi\rangle = a|00\rangle + b|01\rangle + c|10\rangle + d|11\rangle = \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix}$$

- **Tensor products**

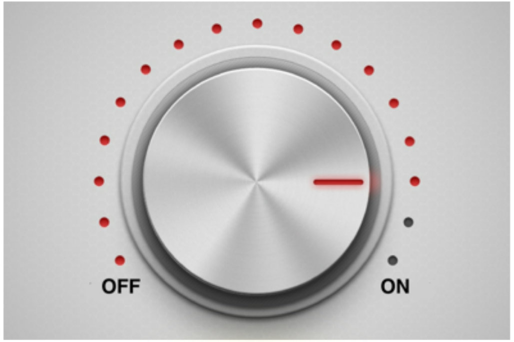
$$|\psi\rangle = (a_1|0\rangle + b_1|1\rangle) \otimes (a_2|0\rangle + b_2|1\rangle) = \begin{bmatrix} a_1 a_2 \\ b_1 a_2 \\ a_1 b_2 \\ b_1 b_2 \end{bmatrix}$$

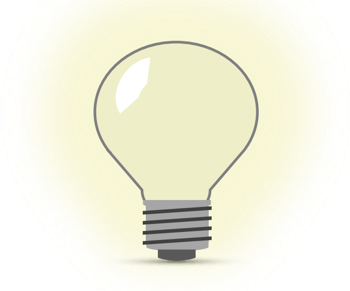
What’s the difference between a classical and quantum bit?



State = OFF

State = ON





State = a*OFF + b*ON

$$|\psi\rangle = a|0\rangle + b\,|1\rangle$$

Qubits are represented on a Bloch Sphere

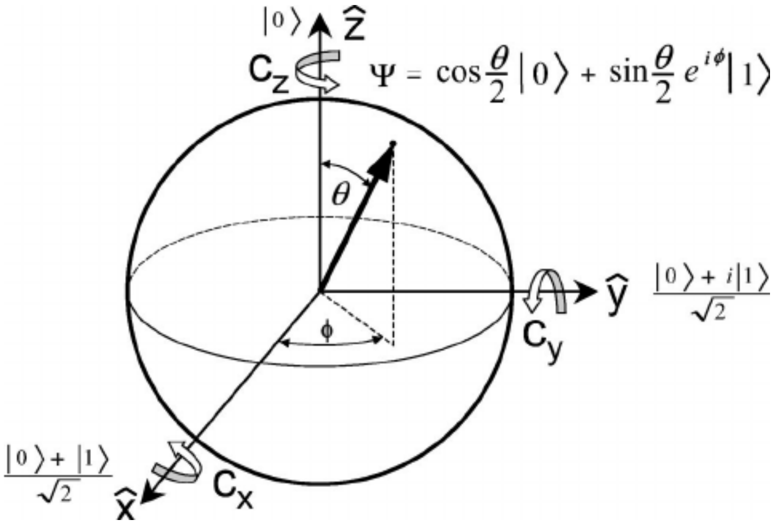
$$|\psi\rangle = a|0\rangle + b\,|1\rangle$$

- **Coefficients *a* and *b* are complex numbers**

  0 with probability $|a|^2$

  1 with probability $|b|^2$

- **So, it's not a probability on a number line**



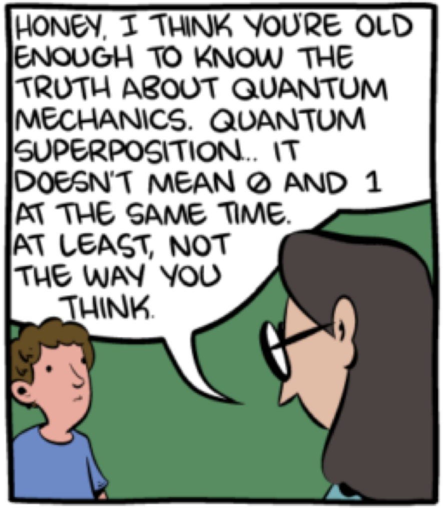
BERKELEY LAB

U.S. DEPARTMENT OF ENERGY

Superposition, or being both in 0 and 1 at the same time…

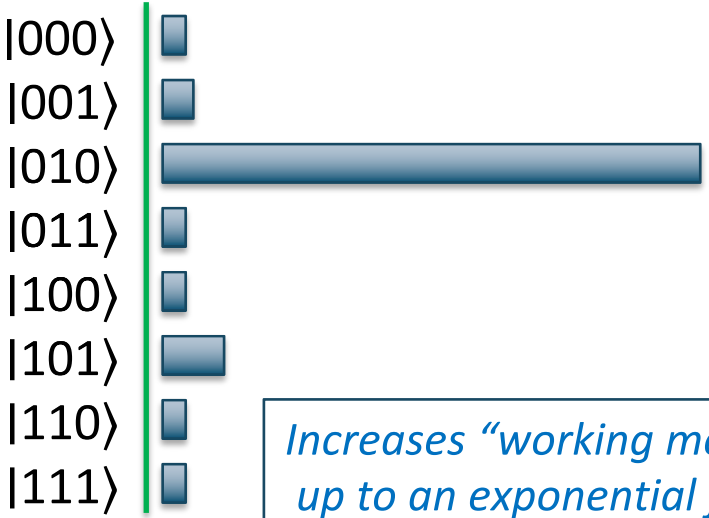
Classical

Bits represents a *single value*, out of 2^N possible bit strings, e.g. 000 == 0



Quantum

Bits represent an *ensemble* of all 2^N possible bit strings, from which you can sample, e.g. for 3 qubits:



Increases "working memory" up to an exponential factor.

Schrödinger's Cat: Dead or Alive

You can only MEASURE either dead or alive, not both

Measuring a quantum bit

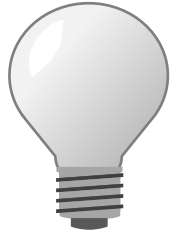
On average

b = 6040 a = 7050

State = a*OFF + b*ON

Measure many
times (sample)

OR



State = ON State = OFF

You only measure either ON or OFF each time with probability equal to a and b squared

Operating on a qubit = Matrix-Vector operations

$$R_x(\phi) = \begin{bmatrix} \cos\left(\frac{\phi}{2}\right) & -i\sin\left(\frac{\phi}{2}\right) \\ -i\sin\left(\frac{\phi}{2}\right) & \cos\left(\frac{\phi}{2}\right) \end{bmatrix}$$

$$R_y(\phi) = \begin{bmatrix} \cos\left(\frac{\phi}{2}\right) & -\sin\left(\frac{\phi}{2}\right) \\ \sin\left(\frac{\phi}{2}\right) & \cos\left(\frac{\phi}{2}\right) \end{bmatrix}$$

$$R_z(\phi) = \begin{bmatrix} e^{-i\frac{\phi}{2}} & 0 \\ 0 & e^{i\frac{\phi}{2}} \end{bmatrix}$$

Rotations around an axis

Pauli-X	$-\boxed{X}-$	$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
Pauli-Y	$-\boxed{Y}-$	$\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$
Pauli-Z	$-\boxed{Z}-$	$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$

Pauli matrices

$$|\psi\rangle = a|0\rangle + b|1\rangle \xrightarrow{X} |\psi\rangle = b|0\rangle + a|1\rangle$$

$$|\psi\rangle = |0\rangle \xrightarrow{H} \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$$

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

Hadamard is special gate sauce

BERKELEY LAB

U.S. DEPARTMENT OF ENERGY

Entanglement, or making qubits interconnected

- **Unifies multiple qubits into a single state**

  Example (*maximum entanglement*):

  $$|\psi\rangle = (|00\rangle + |11\rangle)/\sqrt{2}$$

  $\Longrightarrow$ measuring one qubit determines state of the other

- **A "physical" resource**
  – Can be "added", "removed", used, and quantified (*entanglement entropy*)

- **Allows "instantaneous" operation on all qubits**
  – Popular: with superposition, "try all solutions in parallel"
  – Mathematically: off-diagonal elements in $2^N \times 2^N$ state matrix

Increases information density up to an exponential factor.

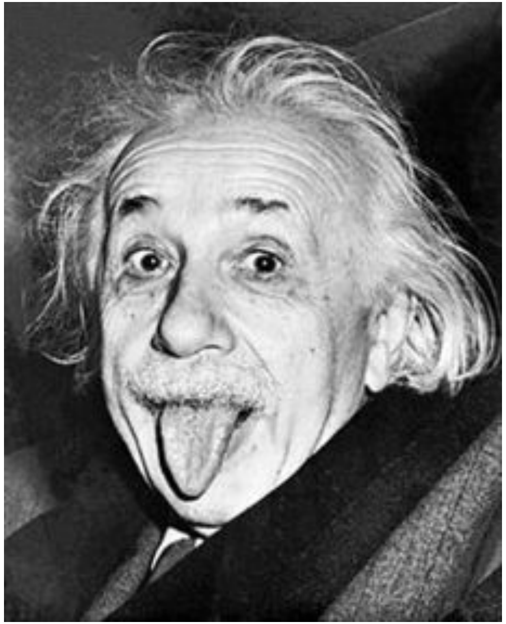
Einstein called it “spooky action at a distance”

EINSTEIN ATTACKS QUANTUM THEORY

Scientist and Two Colleagues Find It Is Not ‘Complete’ Even Though ‘Correct.’

SEE FULLER ONE POSSIBLE

Believe a Whole Description of ‘the Physical Reality’ Can Be Provided Eventually.



Math of entanglement

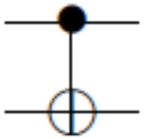
- **Not entangled means you can separate information of qubits**

$$\begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} a_1 a_2 \\ b_1 a_2 \\ a_1 b_2 \\ b_1 b_2 \end{bmatrix} \quad \Longrightarrow \quad ad = bc = a_1 a_2 b_1 b_2$$

- **Effectively you can write the combined state as a tensor product of two Hilbert spaces**
- **If $ad \neq bc$ we call the qubits entangled**

How do we entangle two qubits?

Controlled-NOT (CNOT)



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

- **Change state of second qubit is controlled by first qubit**

$$|00\rangle \Rightarrow |00\rangle$$

$$|01\rangle \Rightarrow |01\rangle$$

$$|10\rangle \Rightarrow |11\rangle$$

$$|11\rangle \Rightarrow |10\rangle$$

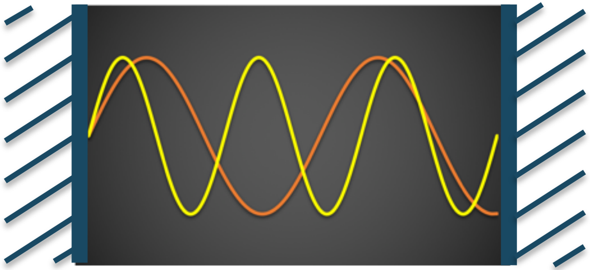
$|AB\rangle \Rightarrow |A(A \oplus B\rangle$ **or addition mod(2)**

Interference

- **Total probability over all bit strings sums to *one***
  - Combined effect of superposition and entanglement

    ⇒ *As one solution becomes more likely (larger amplitude), others have to become less likely (lower amplitude).*

- **Amplify right solution, suppress others**
  - Physics: wave mechanics
  - Popular: music/orchestra
  - Mathematics: complex ($\mathbb{C}$) math

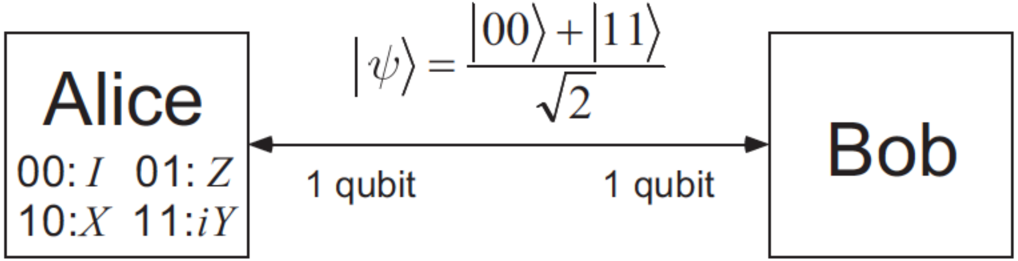


Example, Shor factorization: 3 and 5 fit a whole number of times in 15 ⇒ "standing waves", others interfere destructively.

Interference is how quantum algorithms are designed to work.

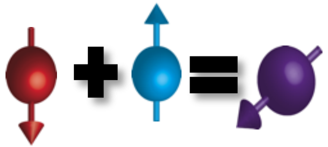
More ”spooky action”, moving 2 bits with 1 qubit

- Moving 2 bits of information with 1 qubit only

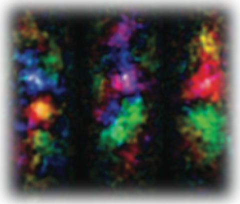


- Bob does a CNOT followed by Hadamard on Alice’s qubit
- Resulting state will be one of fours possible states

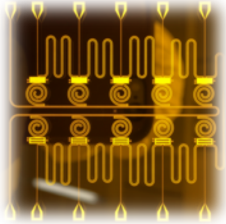
Quantum computing hardware technologies



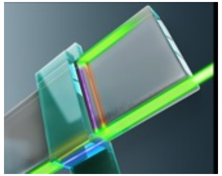
ELECTRONS
SPIN UP + SPIN DOWN



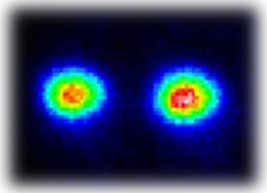
ATOMS



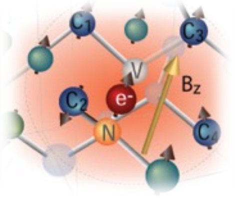
SUPER-
CONDUCTING



MAJORANA
QUASI-PARTICLE

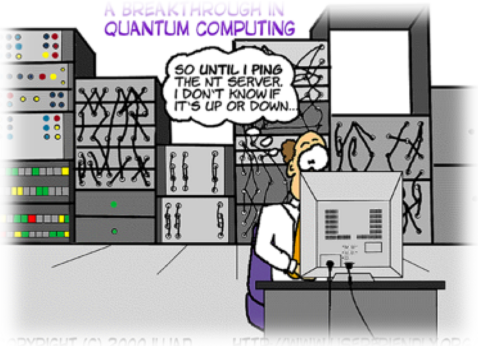


IONS



SOLID STATE
(spins)

D-WAVE



D-Wave The Quantum Computing Company
IBM
Google
rigetti
IONQ
Microsoft
intel

Strongest contenders … at least right now

Superconducting Qubits

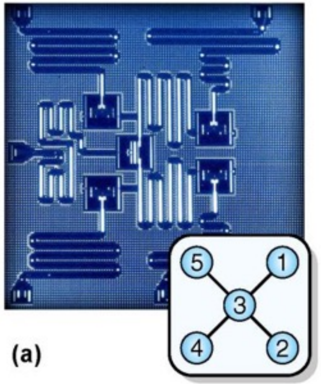
(transmon, flux, phase)

- Qubit –Josephson junctions + capacitors
- Information encoded by superconductor charge
- Controlled by microwave
- Dilution fridge required
- Gates: rotations, CNOT, CZ

Trapped Ion Qubits/Qudits

(hyperfine, optical)

- Qubit – ion (Ca, Yb) trapped in vacuum
- Information encoded in energy levels
- Controlled by laser
- Room temperature
- Gates: Alltoall, Ising, phase shift



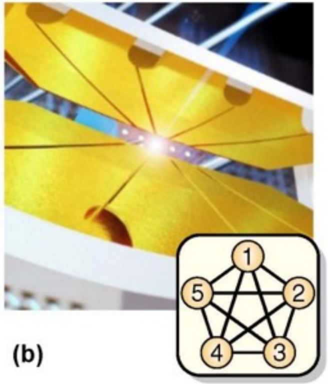
(a)

T2: ~100µs
Gate: ~10ns

T2: ~1s
Gate: ~µs

Commercially viable technologies, fully explored

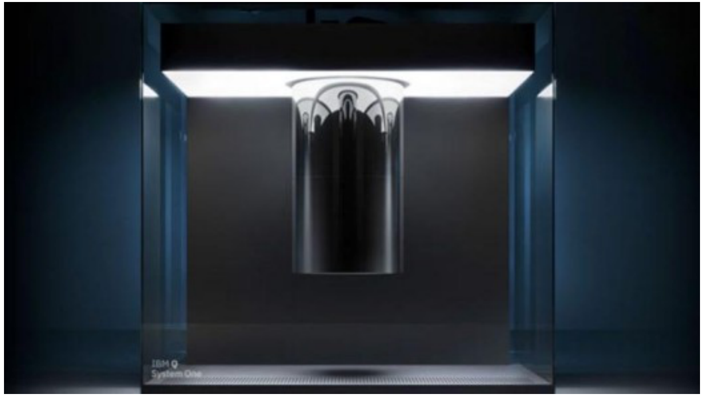
- Superconducting deemed as scalable
- Ions deemed less noisy (T_2), room temp



(b)

What does a SC qubit computer look like?

IBM System One

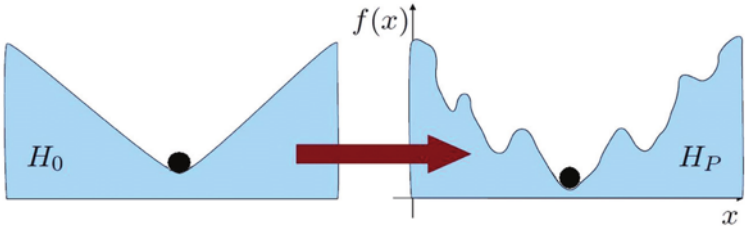


What about the DWave annealer…

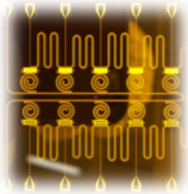
- **In essence superconducting qubits**
  - Adiabatic quantum computer
  - Thousands of bits

- **Debate on quantumness still raging**

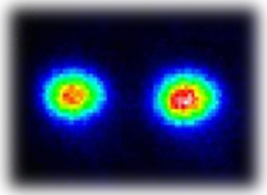
- **Good for very specific problems**
  - Optimization
  - Graph problems



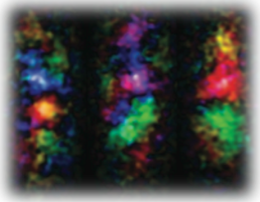
Many challenges with quantum hardware



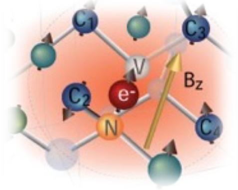
CIRCUITS



IONS



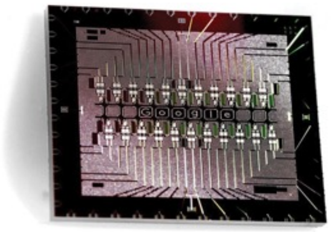
ATOMS

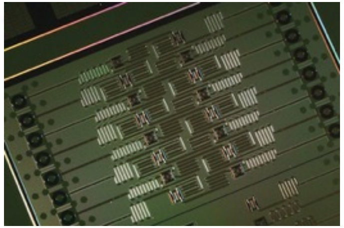


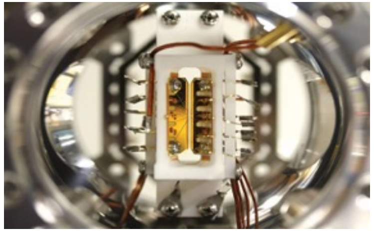
SOLID STATE

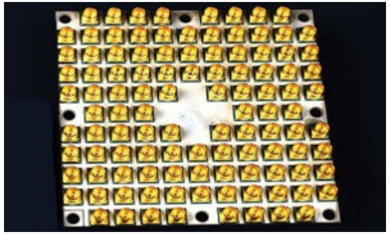
- # of good qubits not yet enough for quantum supremacy/science
- Diverse technologies, each with its own instruction set
- Coherence (available compute time) very short (10s-100s of ops)
- Noise and errors still pretty large

For example, gate sets in superconducting chips


Google

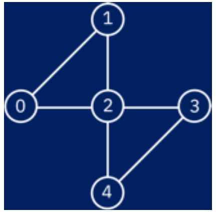

IBM


Rigetti

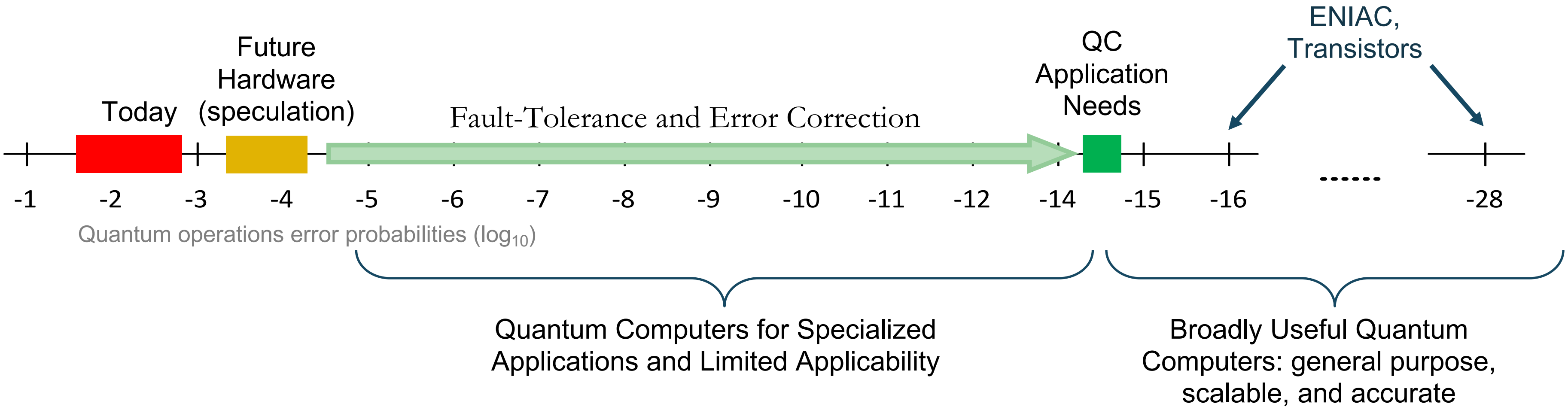

Intel

IonQ

- **Each chip has own native gate set**
  - Single qubit, usually rotations, and Hadamard
  - Two-qubit, usually CNOT, CZ (Google), SWAP
- **Each chip has a constrained topology**
  - Ring, array, mesh, bow-tie
- **Compilers needed to translate gate sets, do mapping**



Noisy intermediate-scale quantum devices



- **Right now quantum computing is still a *physics experiment***
  - Noise is everywhere
  - Measurement errors

Image: SNL, [1] IonQ (https://ionq.co)

Nature v. 555, pages 633–637 (2018)

Nature v. 549, pages 242–246 (2017)

Qubit errors due to relaxation and decoherence

T_1: relaxation, dampening

- Environment exchanges energy with the qubit, mixing the two states by stimulated emission or absorption
- Important during read-out

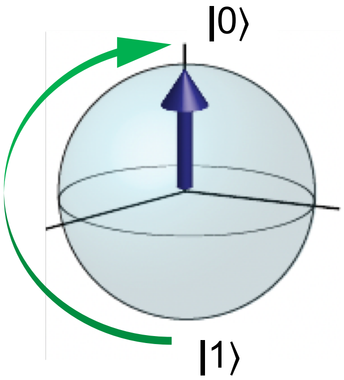
- **Intuitively time to decay from $|1\rangle$ to $|0\rangle$**

T_2: dephasing

- Environment creates loss of phase memory by smearing energy levels, changing phase velocity
- Important during "computation", bounds circuit depth (number of consecutive gates)

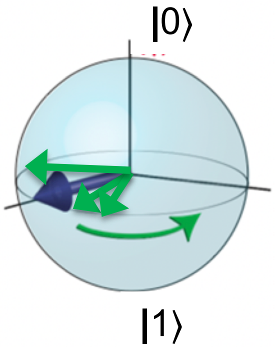
- **Intuitively time for ϕ to get imprecise**

$T_1 > T_2$



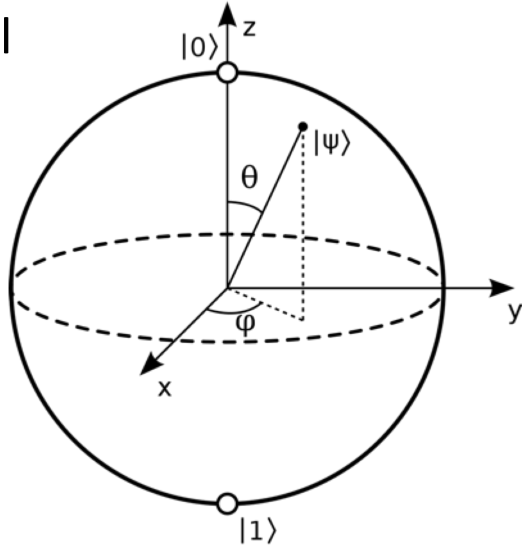
$$|\psi\rangle = \cos(\theta/2)|0\rangle + e^{i\phi}\sin(\theta/2)|1\rangle$$

These are not cut-off times,
but "half-lives."
Decay is *continuous*.



How can we correct for quantum errors?

- **Quantum computing is *analog***
  - Sensitive to noise: no projection to 0 or 1 as in digital
  - *All states are valid*: can not detect noisy results
- **Use group theory: algebra over *logical* qubits**
  - Use multiple qubits to represent states
  - Errors fall outside the group and can be detected
  - Stabilizer codes map errors back onto the group
  - Will require 1000s of qubits: not near-term



States are continuous and all are valid

Example (3-bit flip code):

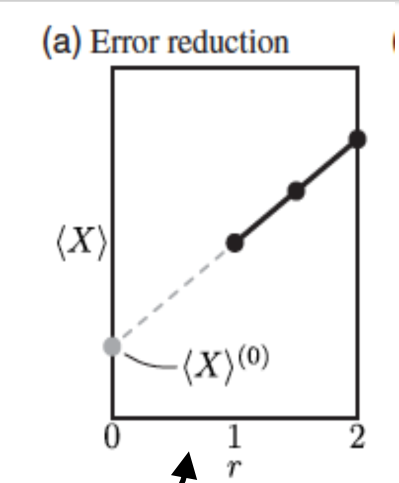
$|0\rangle \rightarrow |0_L\rangle \equiv |000\rangle$

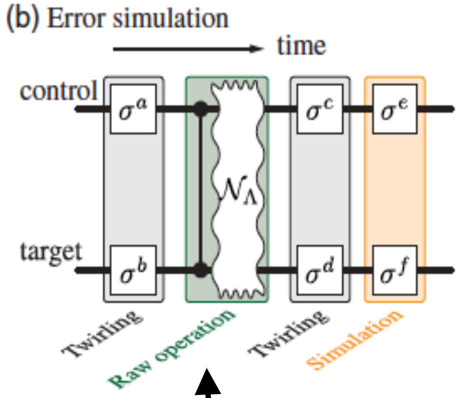
$|1\rangle \rightarrow |1_L\rangle \equiv |111\rangle$

Single bit-flip leads to detectable (and correctable) state:

$|101\rangle \rightarrow |111\rangle$

Reducing stochastic noise in quantum operations





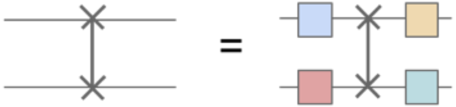
Adding error on purpose

Converting non-stochastic to stochastic (randomized benchmarking)

Concept

- **Add gates that randomize signs of errors each run so *average* result is correct:**

- Implementation: Sandwich “hard” gates between certain random “easy” gates



Ying Li and Simon C. Benjamin - Phys. Rev. X 7, 021050 (2017)

BERKELEY LAB

U.S. DEPARTMENT OF ENERGY

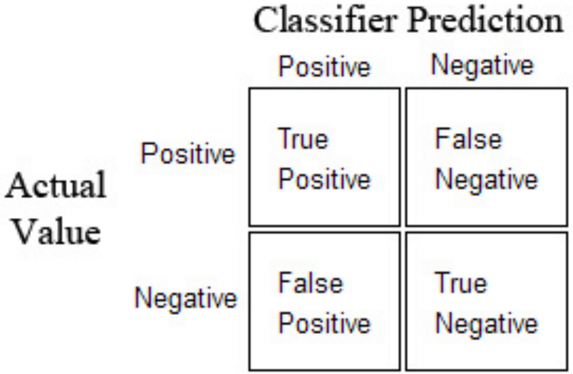
Quick and dirty on correcting of measurement errors

One qubit measurement (IBMQX4):

|0> {'00000': 7904, '00001': 197, '00010': 85, '00011': 6}
|1> {'00000': 800, '00001': 7285, '00010': 11, '00011': 96}

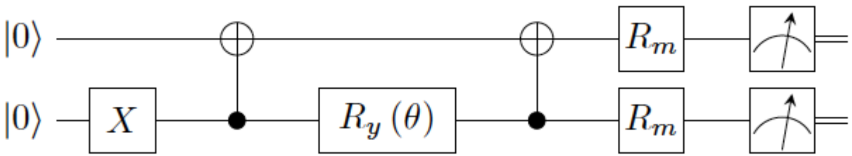
Two qubit measurement

|00> {'00000': 7909, '00001': 191, '00010': 89, '00011': 3}
|01> {'00000': 707, '00001': 7382, '00010': 8, '00011': 95}
|10> {'00000': 585, '00001': 19, '00010': 7409, '00011': 179}
|11> {'00000': 66, '00001': 507, '00010': 686, '00011': 6933}



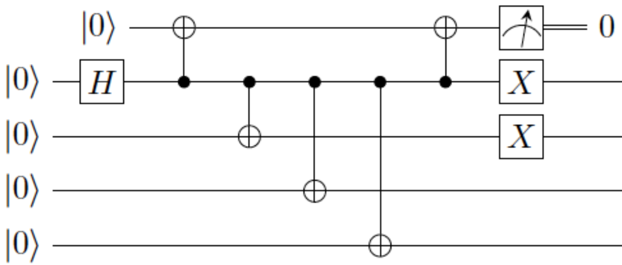
Correction with covariance matrices, disentangling confusion

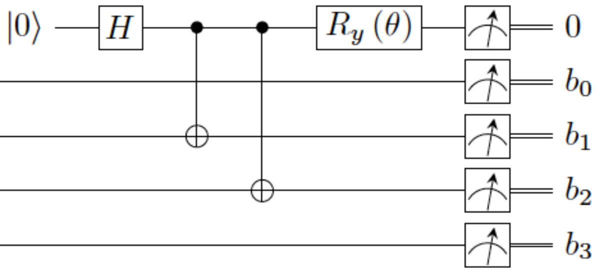
We can also build error detection/correction into circuits



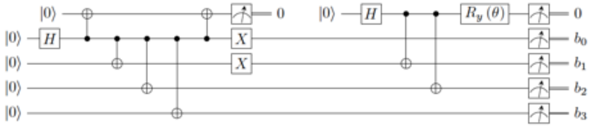
Single error detection

Magic states





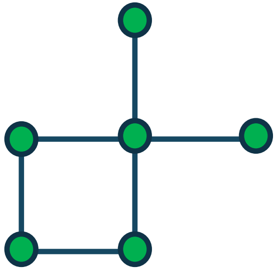
All qubits are equal, but some qubits are more equal than others

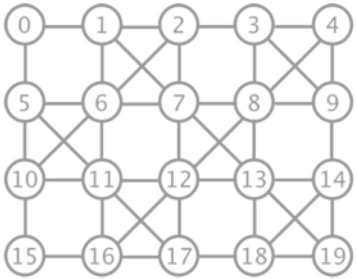


Circuit

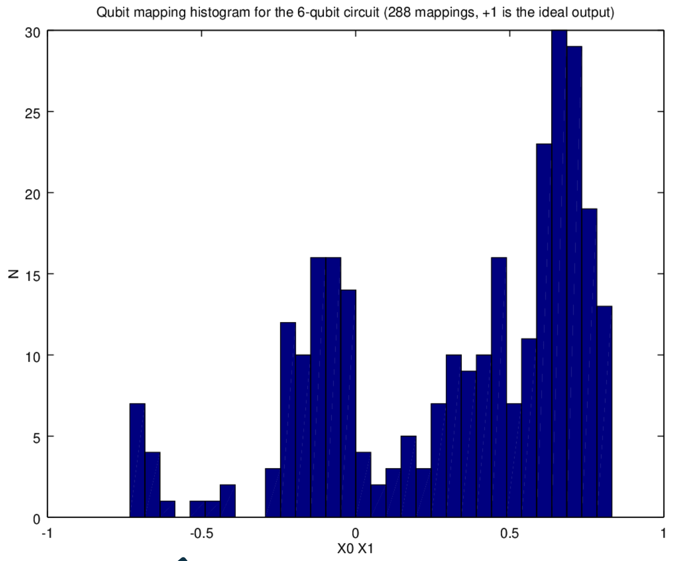
+

Topology

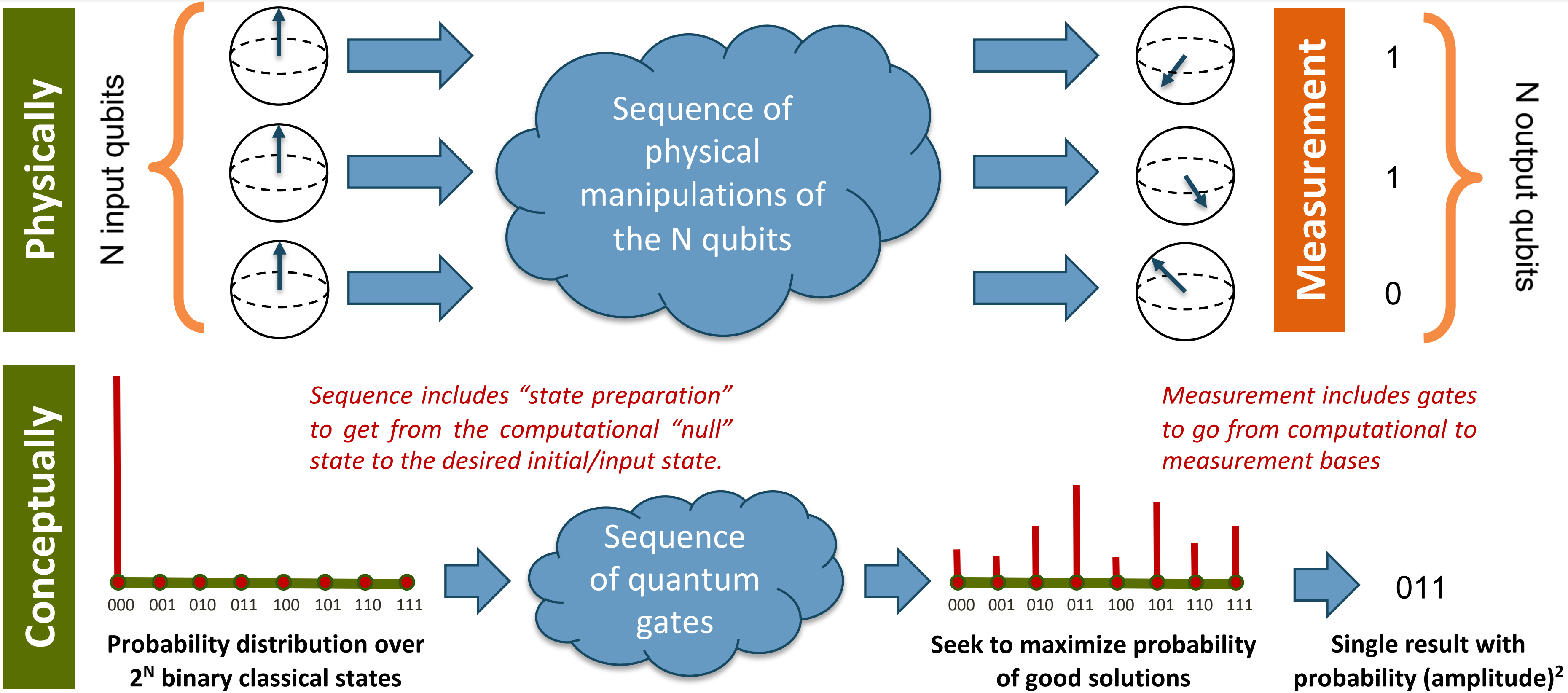




IBMQ Tokyo Hardware

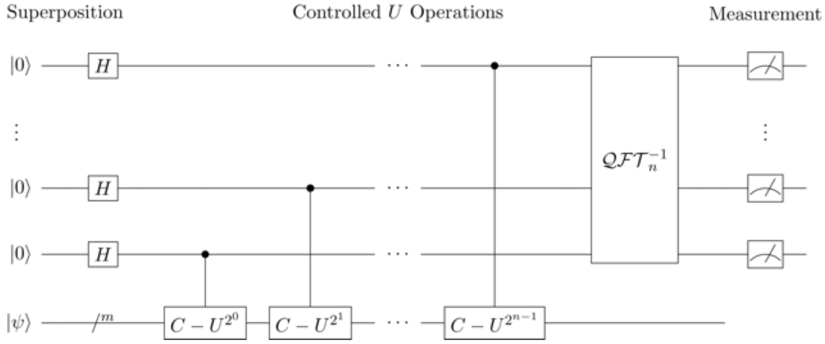


What does a quantum algorithm look like?

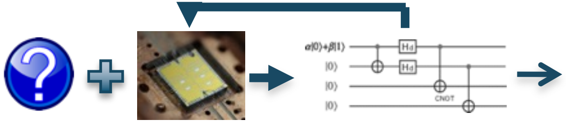


Two common algorithms for quantum simulations

Quantum Phase Estimation (QPE)

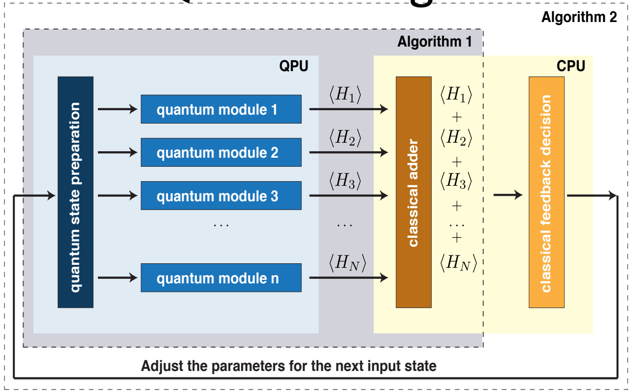


Prepare, evolve, FT and measure to find eigenvalue for eigenvector



42.02

Variational Quantum Eigensolver (VQE)



$$H = \sum_{i\alpha} g_i^{\alpha} \langle \sigma_\alpha^i \rangle + \frac{1}{2} \sum_{ij\alpha\beta} g_{ij}^{\alpha\beta} \langle \sigma_\alpha^i \sigma_\beta^j \rangle + \cdots$$

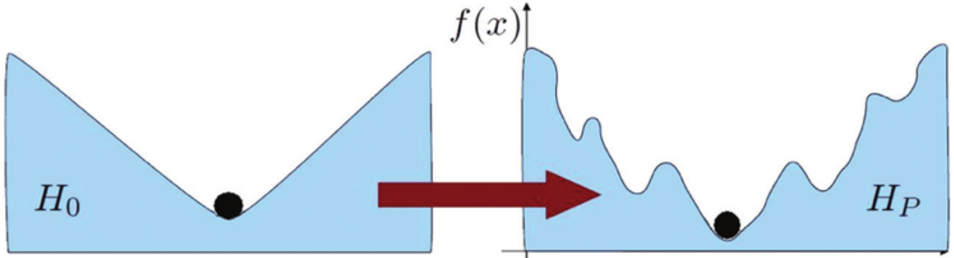
Only prepare and measure, do the rest classically

Adiabatic quantum computing algorithm

Put quantum system in lowest-energy configuration in a way that's easy to do

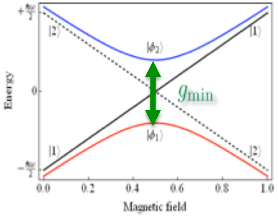
$$H_0 = \sum_{i=1}^{n} \sigma_x^{(i)} \qquad |\psi_0\rangle = \frac{1}{2^{n/2}} \sum_{i=1}^{2^n} |i\rangle$$

Evolve the quantum system in a way that keeps it in its lowest-energy configuration throughout



$$H(s) = (1 - s)H_0 + sH_1$$

Readout success of final state most probable for evolutions that are close to "adiabatic"



$$H_1 = h_0\, I + \sum_{i=1}^{n} h_i\, \sigma_z^{(i)} + \sum_{i,j=1}^{n} J_{ij}\, \sigma_z^{(i)} \otimes \sigma_z^{(j)}$$

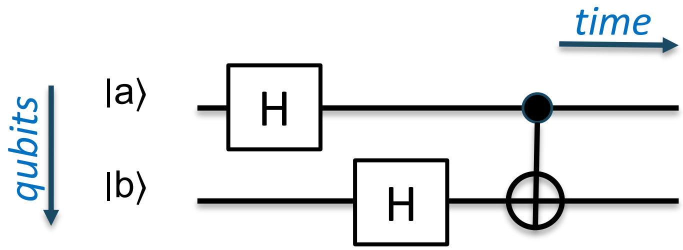
How do we program quantum computers?

Circuit Model

- Diagrams of wires (qubits) and gates (logical operations, applied in order
- Write by hand or generated with science domain software (eg. OpenFermion)
- Hard to generate optimally

Unitary Linear Algebra

- Matrices (operations) and vectors (state)
- Often more natural to science domain (eg. coupling strengths)
- Hard to decompose: 2^N x 2^N in size, with N the number of qubits



$$\Leftrightarrow \quad \frac{1}{2} \times \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 \end{bmatrix} \times \begin{bmatrix} a_1 b_1 \\ a_1 b_2 \\ a_2 b_1 \\ a_2 b_2 \end{bmatrix}$$

Representations are equivalent, can go back and forth, and even mix.

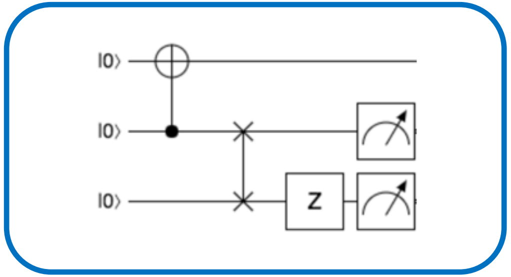
BERKELEY LAB

U.S. DEPARTMENT OF ENERGY

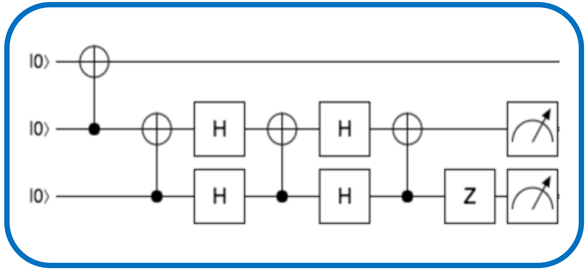
A whole system software stack is needed

Scientist ⟷ *Hardware*

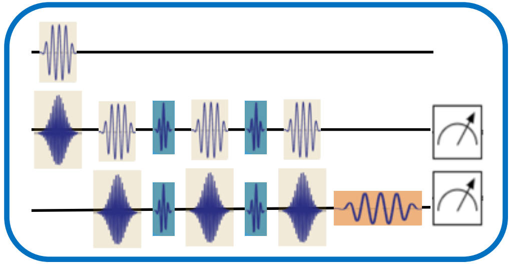
Initial Quantum Algorithm



Compiled Quantum Algorithm



Pulses output by AWG



High level interface

- Arbitrary gates, qubit reset, feedback, measurement
- Algorithm specified in any gate set

Translate to processor

- Arbitrary gates compiled into available gate set
- Processor connectivity and timing constraints enforced

Translation to hardware

- Define pulse parameters (shape, phase, sequence)
- Reset/feedback code applied by FPGAs

Courtesy of Irfan Siddiqi

An incomplete list of software tools

- **Frameworks from most chip providers**

Provider	*Framework*	*License*	*Cloud*
IBM	QisKit	Minor restrictions	IBM Q-Experience
Google	Cirq	Open	
Rigetti	Forest / PyQuil	Restrictive	Rigetti QCS (beta)
Microsoft	LiQUi> / Q#	Minor restrictions	
D-Wave	qbsolv	Minor restrictions	D-Wave Leap

- **Academia & startups target the above**
  - E.g. PyTKET (Cambridge Quantum), ProjectQ (ETH Zürich)
  - QuTiP (Academia, also RIKEN; http://qutip.org)

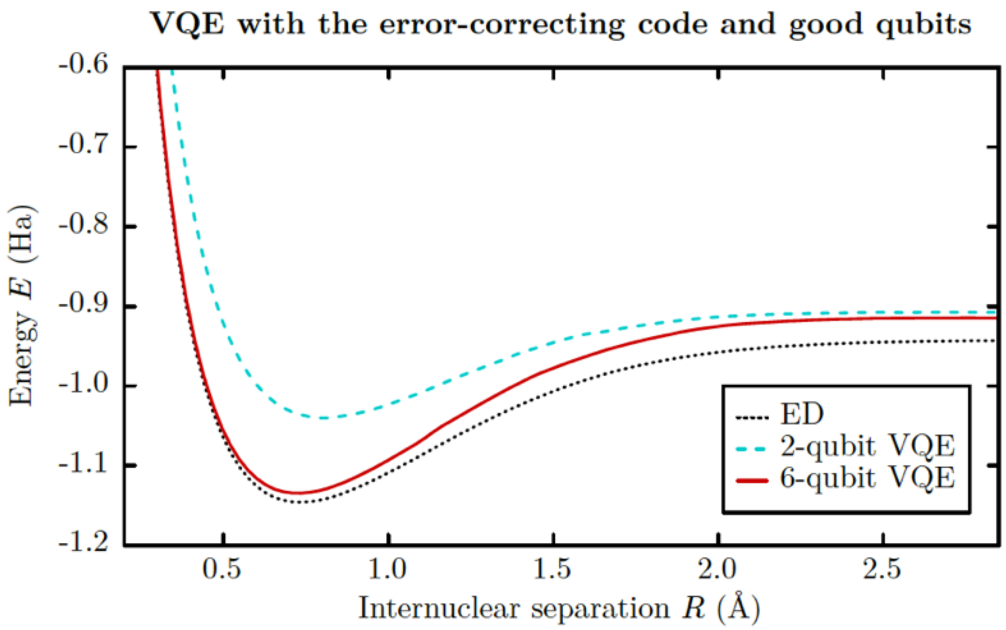
And if you know Python, it’s not that scary…

```
from qiskit import *
from qiskit.compiler import transpile, assemble

qr = QuantumRegister(3)
cr = ClassicalRegister(3)
circuit = QuantumCircuit(qr, cr)
circuit.x(qr[0])
circuit.cx(qr[0], qr[1])
circuit.measure(qr, cr)

qobj = assemble(transpile(circuit, backend=backend), shots=1024)
job = backend.run(qobj)
counts = job.result().get_counts()
print(counts)
```

How good is a quantum computer?: Let's look at H_2



H_2 molecule on 2 qubits with minimal basis

Towards useful quantum computing for science

Hardware technology

Scientific algorithms and software

- Increasing qubit count
- Increasing lifetimes
- Increasing fidelity and reducing errors

- Reducing qubit count
- Decreasing operation counts
- Incorporating error resiliency

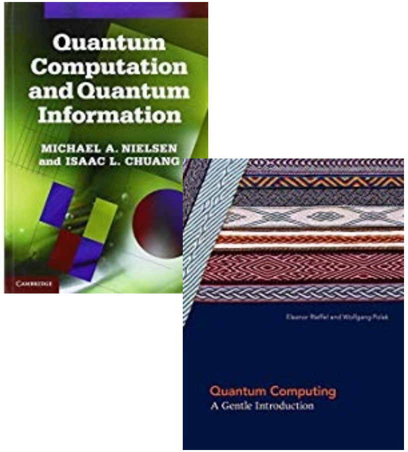
Acknowledgements

This work was supported by the DOE Office of Advanced Scientific Computing Research (ASCR) through the Quantum Algorithm Team and Quantum Testbed Pathfinder Program

This research used computing resources of the Oak Ridge Leadership Computing Facility through the INCITE program and the National Energy Research Scientific Computing Center

Study Resources

- **Nielsen & Chuang "Quantum Computation and Quantum Information"**
  - Complete, lots of material, better for physicists
- **Nielsen's "Quantum computing for the Determined"**
  https://www.youtube.com/playlist?list=PL1826E60FD05B44E4
- **Rieffel & Polak, "A Gentle Introduction"**
  - Targeted at computer scientists and mathematicians
- **John's Preskill's lecture notes**
  http://www.theory.caltech.edu/~preskill/ph219/ph219_2017
- **Todd Brun's lecture notes (insightful)**
  https://www-bcf.usc.edu/~tbrun/Course/
- **Interactive circuit simulator**
  http://algassert.com/quirk



Conferences: http://quantum.info/conf/2019.html
Papers: https://arxiv.org