Intro to Quantum Programming

Matt Norby
Senior Consultant, Manifest Solutions
norbymatt@gmail.com
Twitter: @matt_norby

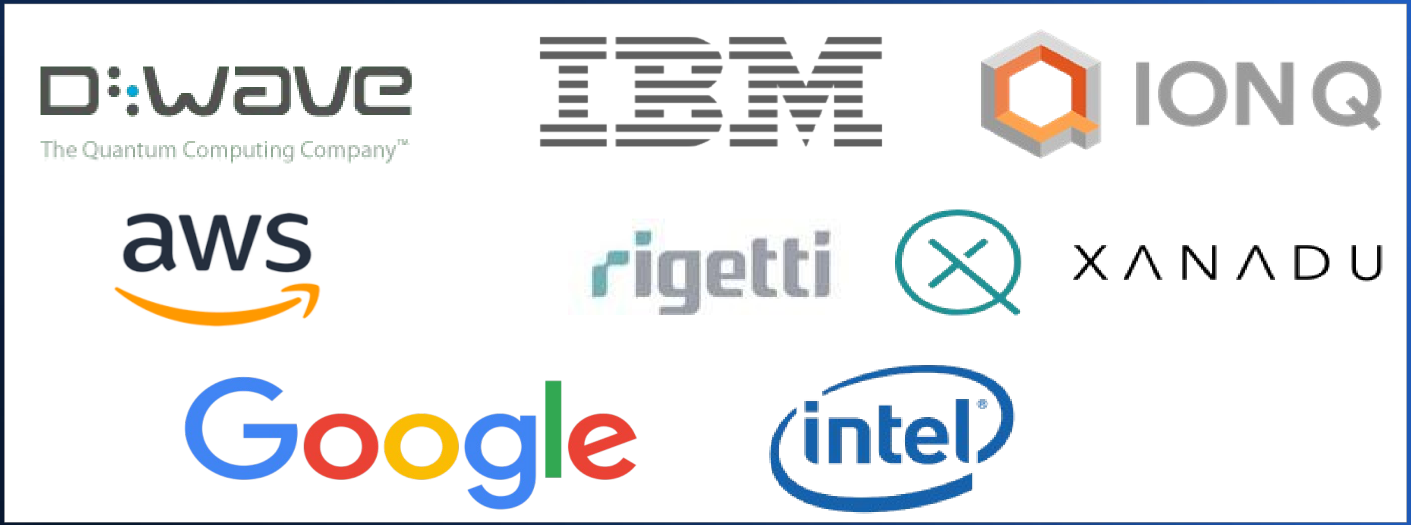
Slide deck available at:
https://github.com/mattnorby/Intro-to-Quantum-Programming

Agenda

- Why quantum computing?
- Quantum physics background and terminology
- A useful model for quantum computing
- Quantum algorithms
- Demo

Why Quantum Computing?

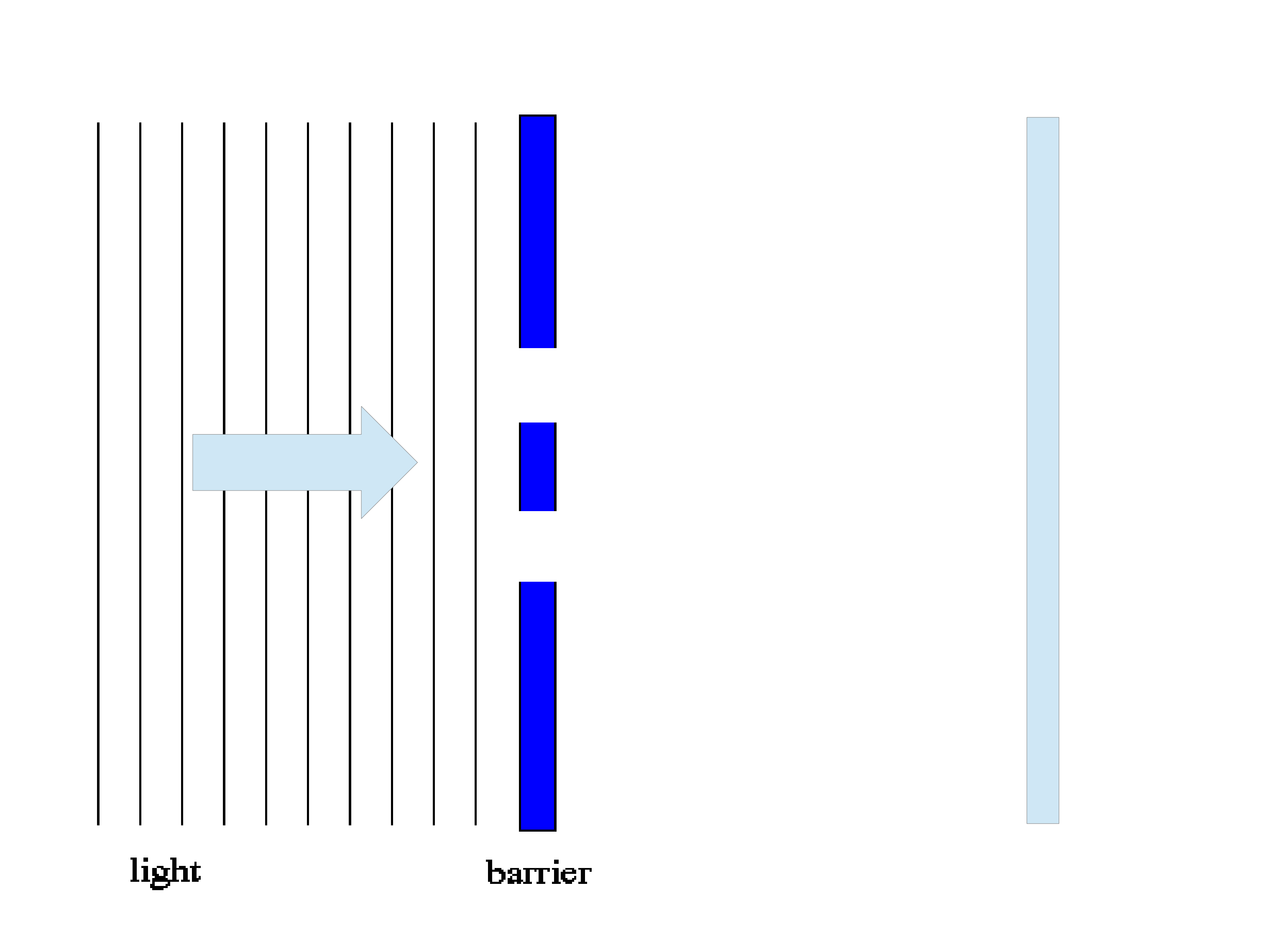
- Much faster algorithms (sometimes)
- Quantum computers exist today
- Computer industry investment



What do we mean by quantum?

- The smallest possible things
  - Indivisible, countable
- Physical effects below atomic scale
  - Very different from macro-scale physics
  - Wave-particle duality

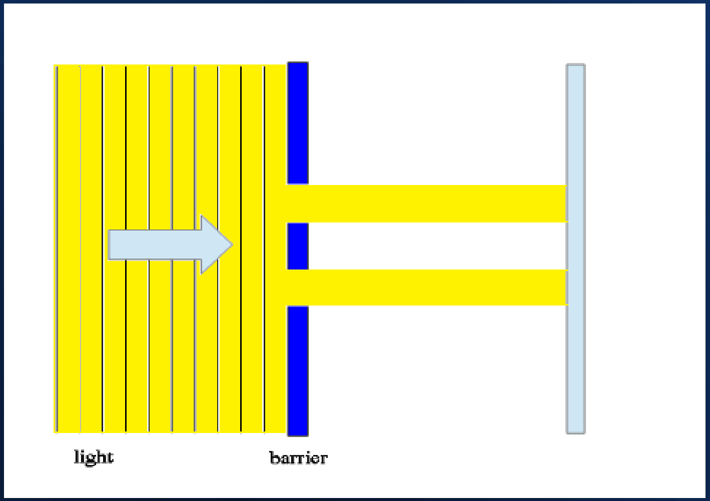
Thomas Young experiment (1801)



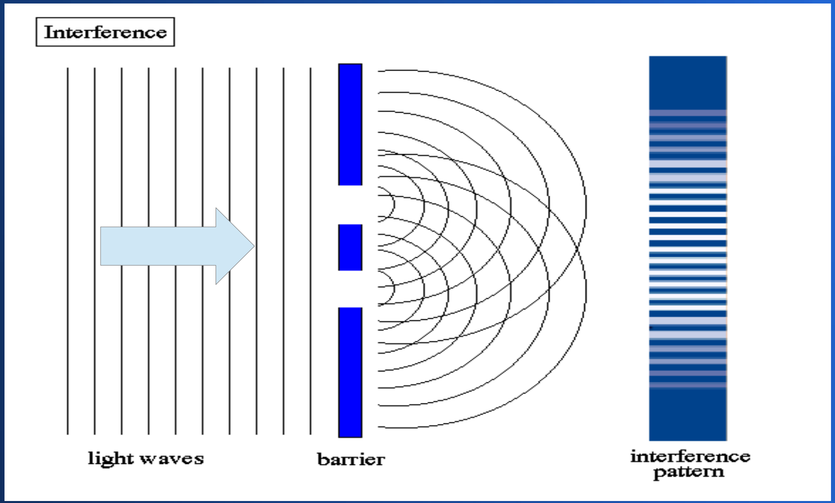
Thomas Young experiment (1801)

The first “double slit” experiment

If light is a particle



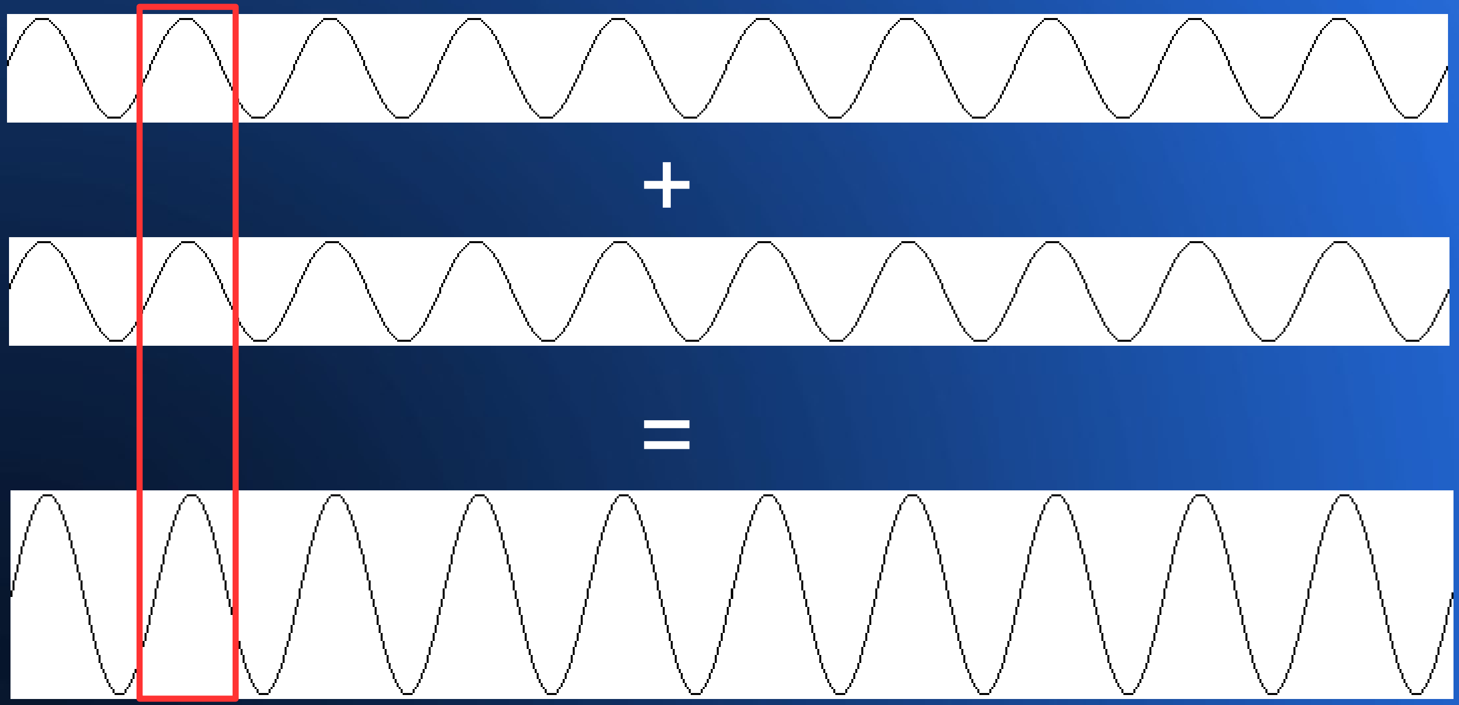
If light is a wave



Interference

When waves collide, they interfere

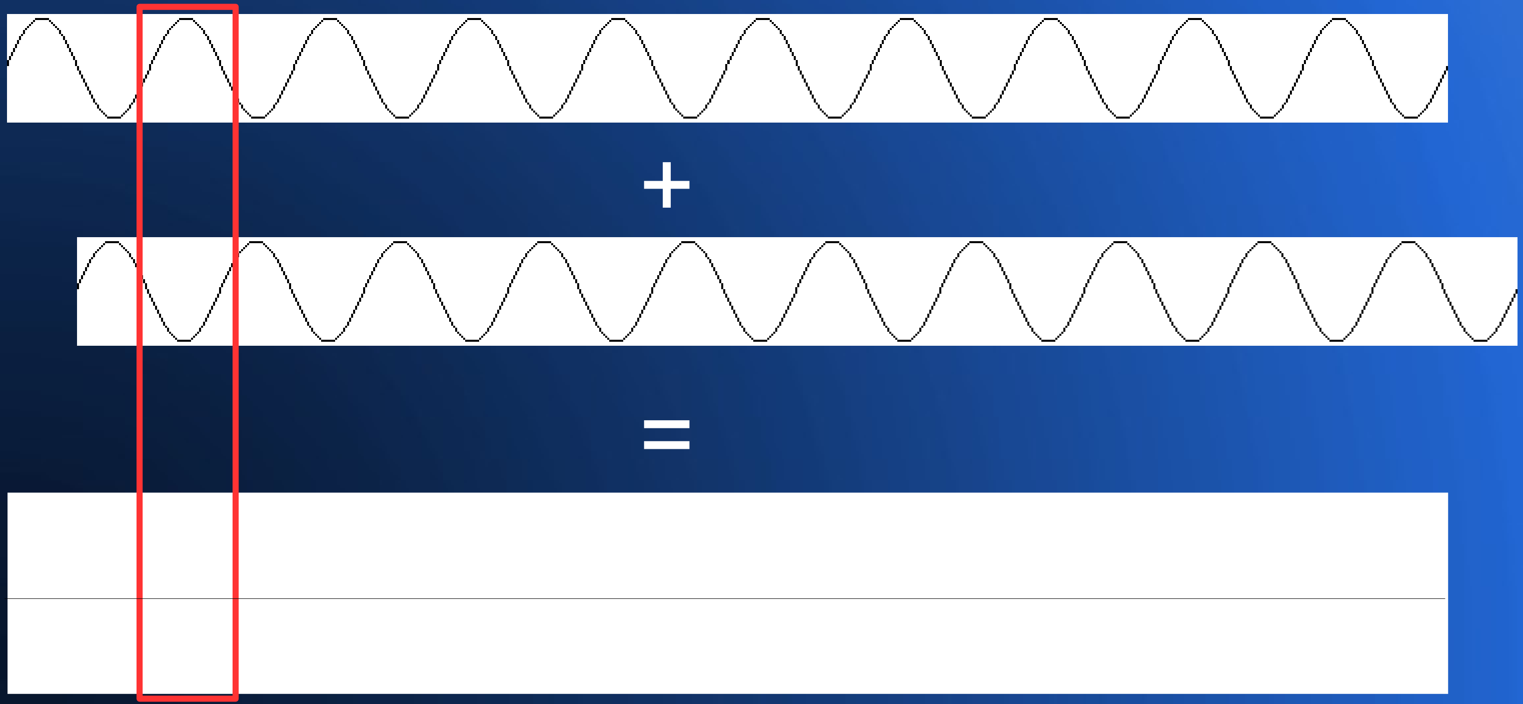
- Peaks meet peaks → constructive interference



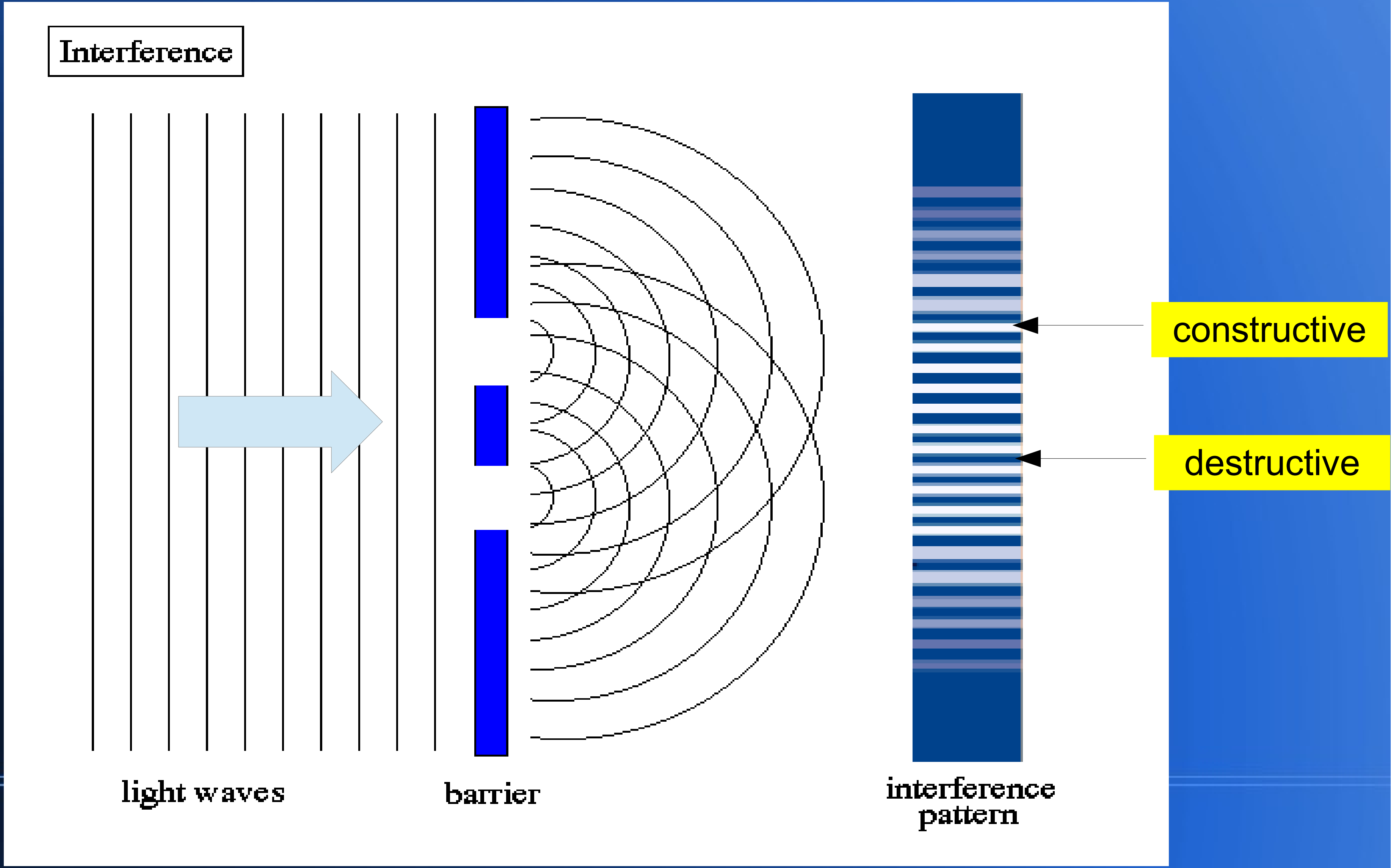
Interference

When waves collide, they interfere

- Peaks meet valleys → destructive interference

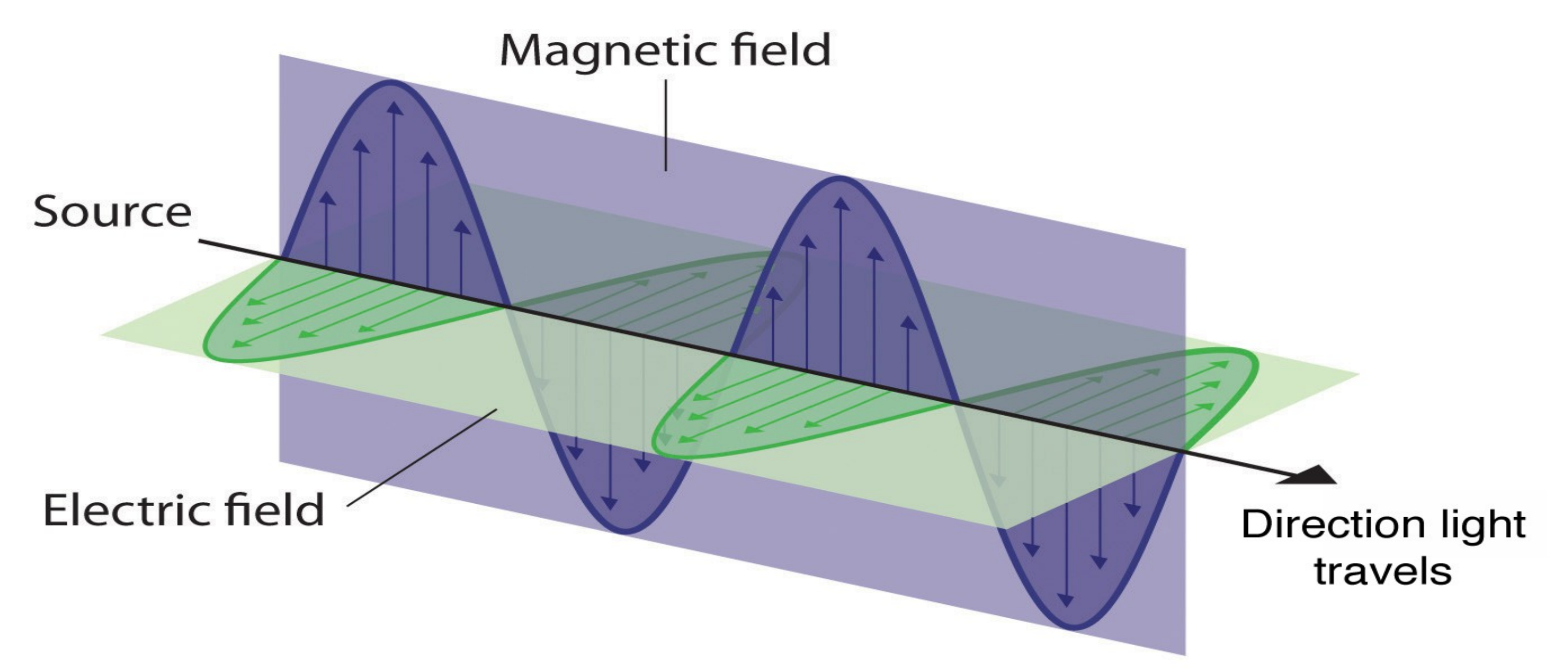


Thomas Young experiment (1801)



James Clerk Maxwell (1860s)

- Light, magnetism, and electricity are related
- Light is an electromagnetic wave
- Light has an orientation



Max Planck (1900)

- Electromagnetic radiation comes in quanta
- Energy is proportional to frequency: E = hv, where h = Planck's constant, v = frequency

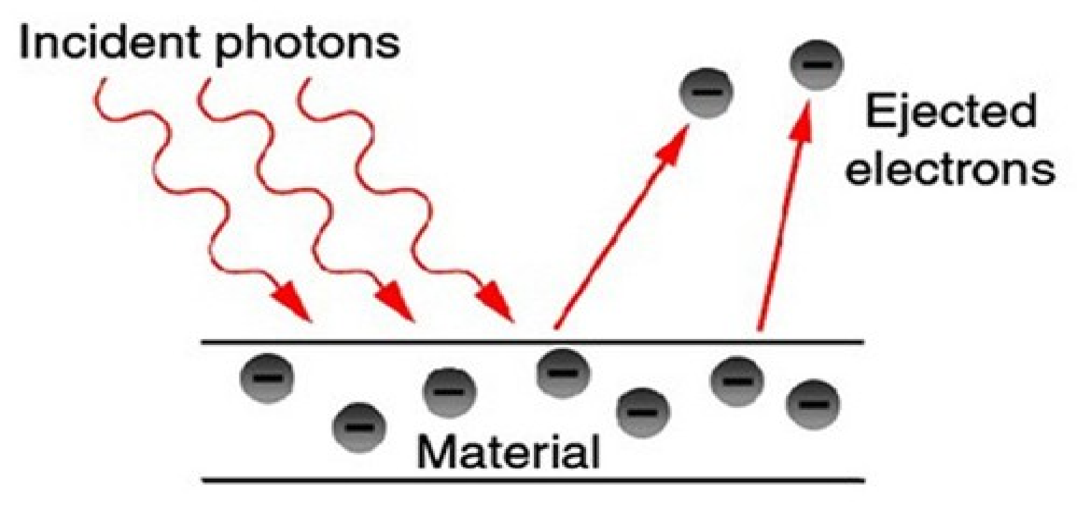
→ Lower limit on light energy.

→ Light exists as discrete “photons” of energy.

Albert Einstein (1905)

Photoelectric effect

- Above a threshold frequency, light causes electrons to be released from metal



→ Light comes in quanta

A qubit is a single quantum of something (e.g. light).

Born / Heisenberg (1925)

Matrix model of quantum mechanics

- Probability of being in a given state
- Imaginary numbers were needed
- Corollary: uncertainty principle
  - Cannot know both position and momentum

The math behind quantum algorithms, which is used to prove that an algorithm will work.

Born / Heisenberg (1925)

Paul Dirac created Dirac notation (“bra/ket”)

→ More compact than matrix representation

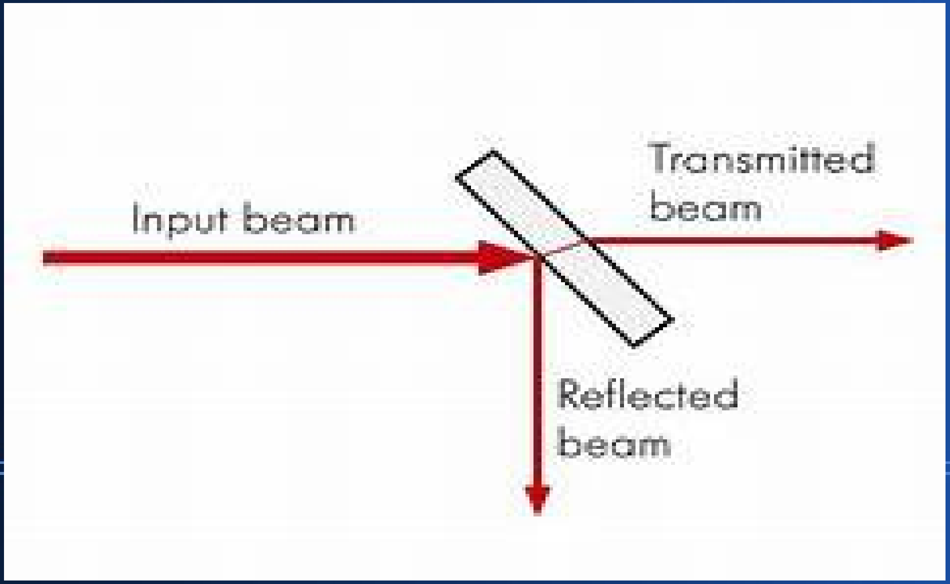
$|0> = \begin{pmatrix}0\\1\end{pmatrix}$	$|1> = \begin{pmatrix}1\\0\end{pmatrix}$	$|01> = \begin{pmatrix}0\\1\\1\\0\end{pmatrix}$
$<0| = \begin{pmatrix}0 & 1\end{pmatrix}$	$<0|0> = \begin{pmatrix}0 & 1\end{pmatrix}\begin{pmatrix}0\\1\end{pmatrix} = 1$	

The math is most often written in Dirac notation.
The “ket” appears in quantum program diagrams.

Alain Aspect (1972)

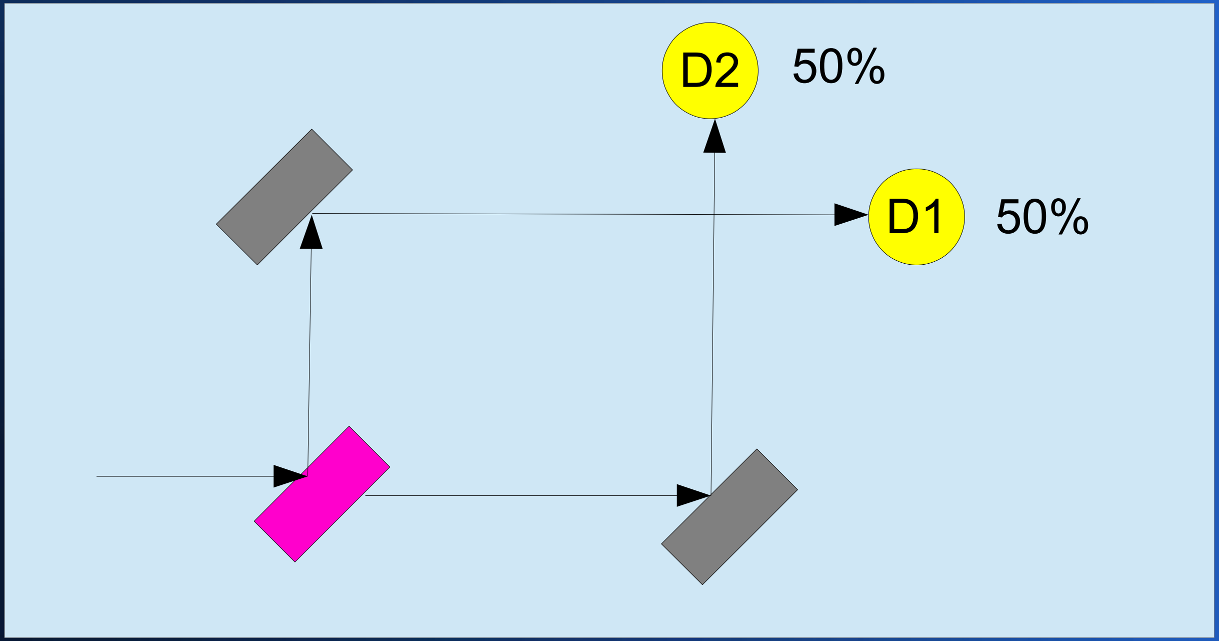
- If you fire one photon at a beamsplitter, it seems to go one way or the other (not both)
- Equal probability for each path

→ Particle behavior with one beamsplitter



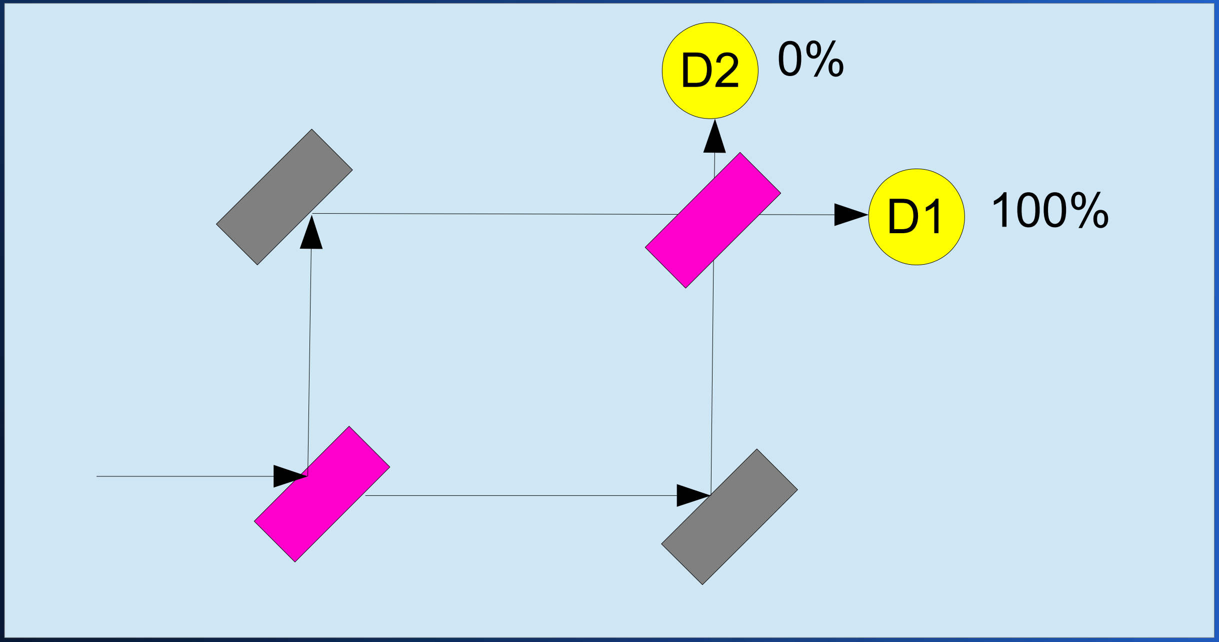
Alain Aspect (1972)

- If you fire one photon at a beamsplitter, and reflect the paths so that they cross, the photon still goes one way or the other



A. Aspect / R. Grangier (1986)

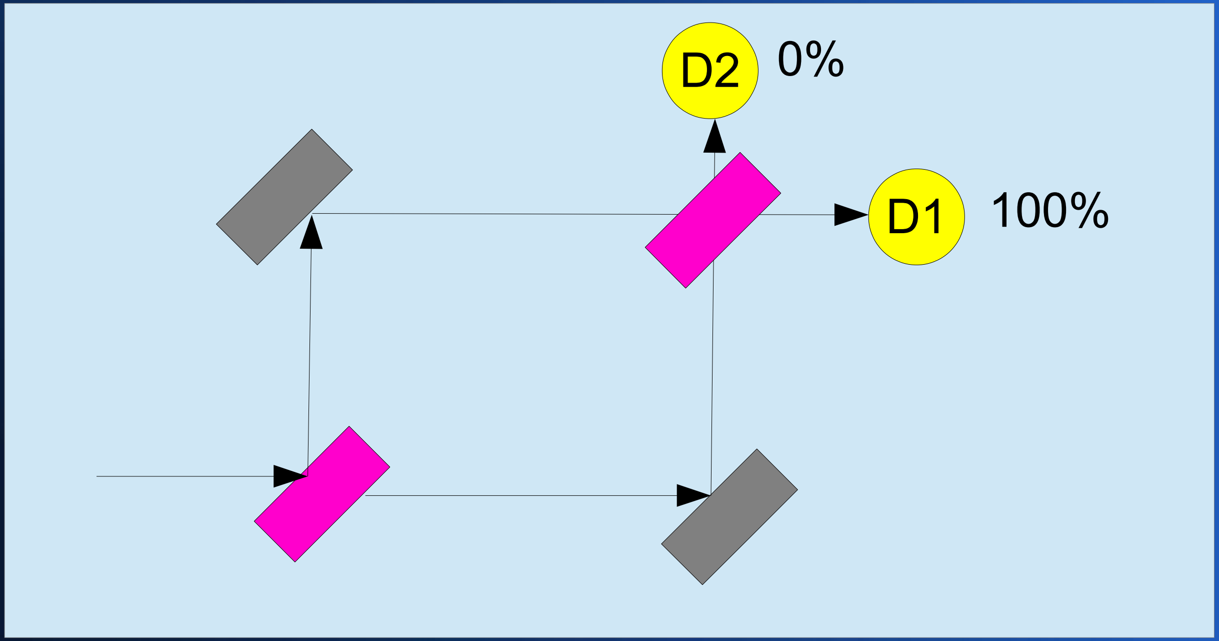
- If you reflect equal-length paths so that they cross at a second beamsplitter, the photon follows *one path* after the second beamsplitter



A. Aspect / R. Grangier (1986)

→ Photon is in superposition when it leaves the first beamsplitter (2 simultaneous states)

→ Photon *interferes with itself* at 2nd beamsplitter!



Superposition

- A particle in superposition of two states represents both states at one time
- Probability of observing one state or the other
- Once the state has been observed, probability no longer applies (observation does not change)

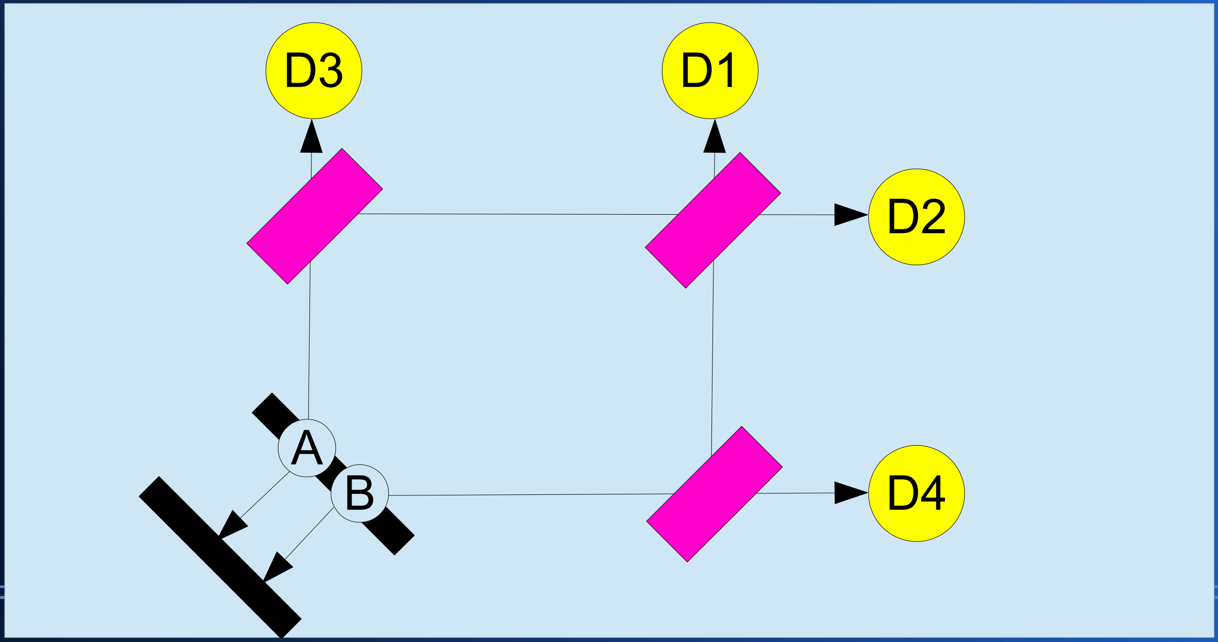
Entanglement

- Entangled particles are in a state that cannot be described separately for each particle
- Measuring one particle determines the measurement for the other
  - same or opposite
- Does not matter which is measured first
- Does not matter how far apart they are

A. Zeilinger (1995)

“Quantum eraser” experiment

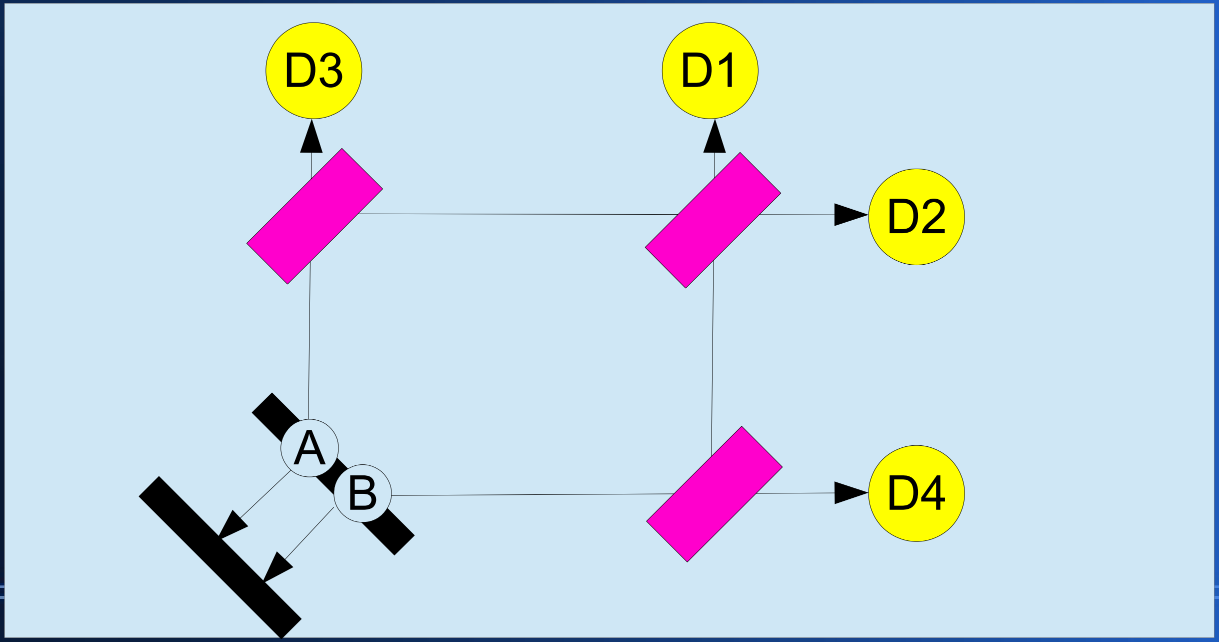
- Entangled “system” and “environment” photons
- Collect or discard “which way” information



A. Zeilinger (1995)

→ When “which way” info is known (D3/D4), there is no interference pattern on the screen

When not known, there is interference (D1/D2)

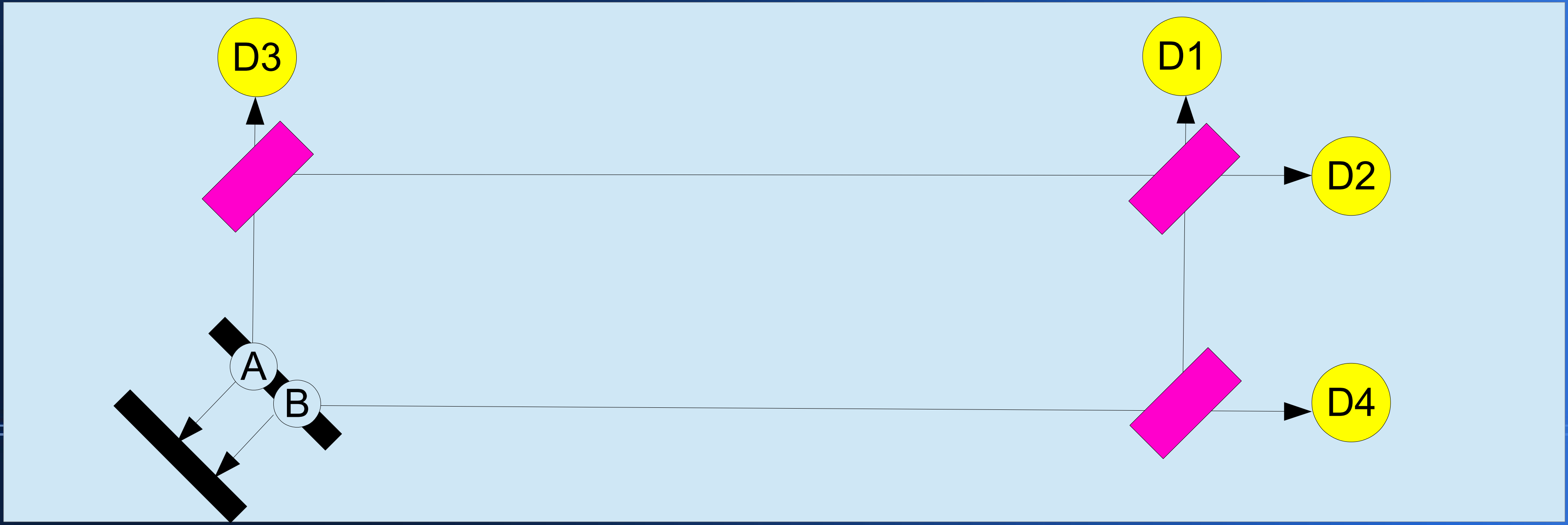


Yoon-Ho Kim et al. (1999)

“Delayed choice quantum eraser” experiment

- Hit the screen *before* the “which way” decision

→ When “which way” info is known, there is no interference pattern; when not known, there is (!)



Quantum Programming

Let's harness this behavior in a computer!

- Qubits
- Superposition
- Entanglement
- Interference
- Quantum gates

Qubits

- Represent a value of zero or one, or a "superposition" of both zero and one
- Manipulated via quantum gates
- Most commercially available quantum computers contain a few dozen qubits
  - D-Wave: 2048 qubits
  - D-Wave Pegasus: > 5000 qubits

Key features of qubits

- Orientation that can be measured
- Behaves like a particle (predictable path)
- Behaves like a wave (interference)
- One qubit can be “entangled” with another
- Capable of being in multiple states at once (superposition)

With N qubits, you can represent 2^N inputs.

Examples of qubits

- Photon of light
  - Measure: polarization of light
- Electron (trapped in a “quantum dot”)
  - Measure: electron spin
- Ion
  - Measure: energy levels

Superposition

- In quantum computing, a qubit in superposition represents both zero and one at the same time
- Probability of *measuring* zero or one
- Measuring the qubit ends superposition

Use superposition to try 0 and 1 at the same time.

Interference

Interference and quantum computing

- Programs will evaluate many inputs at once
- Constructively boost correct outputs
- Destructively reduce incorrect outputs

Use interference to select the correct input.

Entanglement

- Entangled qubits measure the *same* value or *opposite* values
- Measuring one qubit's value determines the other qubit's value when it is measured
- Set up with Hadamard gate + CNOT gate

Use entanglement to relate values across qubits, and reduce possible outputs.

Quantum Programming

Overall Strategy

- Place all data qubits in superposition
- Entangle some of them
- Manipulate with quantum gates
- Use interference to emphasize correct outputs, de-emphasize incorrect outputs
- Measure the data qubits

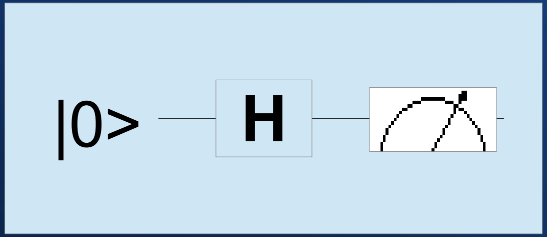
Quantum Gates

- Hadamard (superposition, one qubit)
- X (like a NOT)
- Swap
- CNOT (controlled NOT – entangle qubits)
- Toffoli

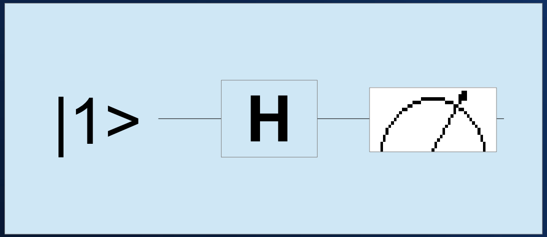
All quantum gates are their own inverses.

Hadamard (H) Gate

Places a qubit in superposition



50% chance to return 1 or 0

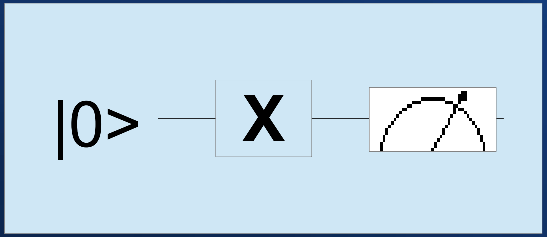


50% chance to return 1 or 0

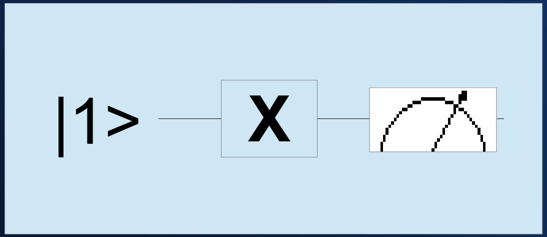
These are the quantum equivalent of “Hello World”.

X Gate

Inverts the state of a qubit



Result is 1

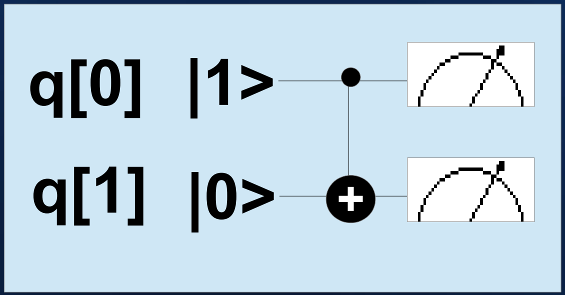


Result is 0

CNOT Gate

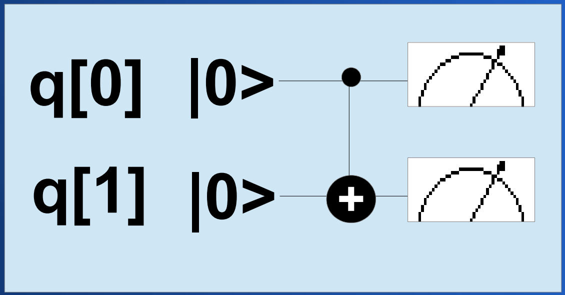
Given 2 qubits, flip the state of the 2nd if the first is 1 (no change otherwise)

Use together with H to entangle qubits



Result is 1

Result is 1

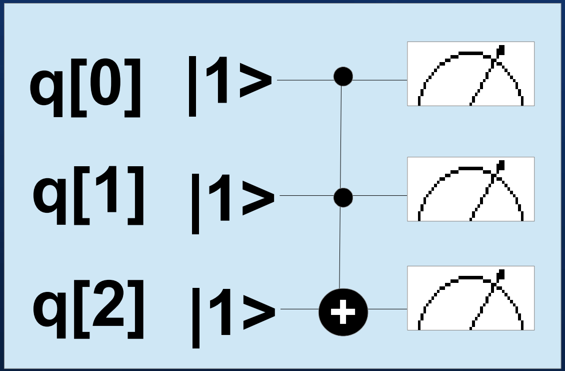


Result is 0

Result is 0

Toffoli Gate

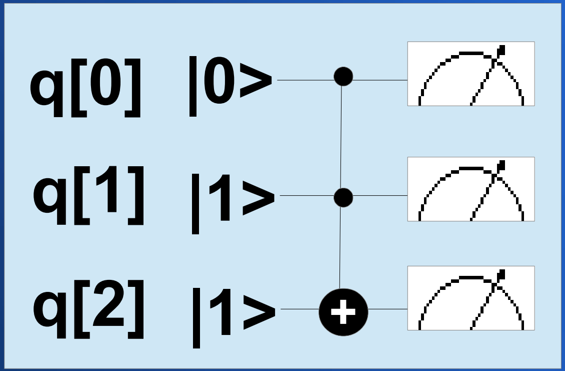
Given 3 qubits, flip the state of the 3rd if the first 2 are 1 (no change otherwise)



Result is 1

Result is 1

Result is 0



Result is 0

Result is 1

Result is 1

Grover's Algorithm

Given an unindexed collection of data, find one specific item among N elements

Assume the data is not indexed or sorted

- Conventional computer: O(N)
- Quantum computer: O(sqrt(N))

Grover's algorithm has been proven optimal

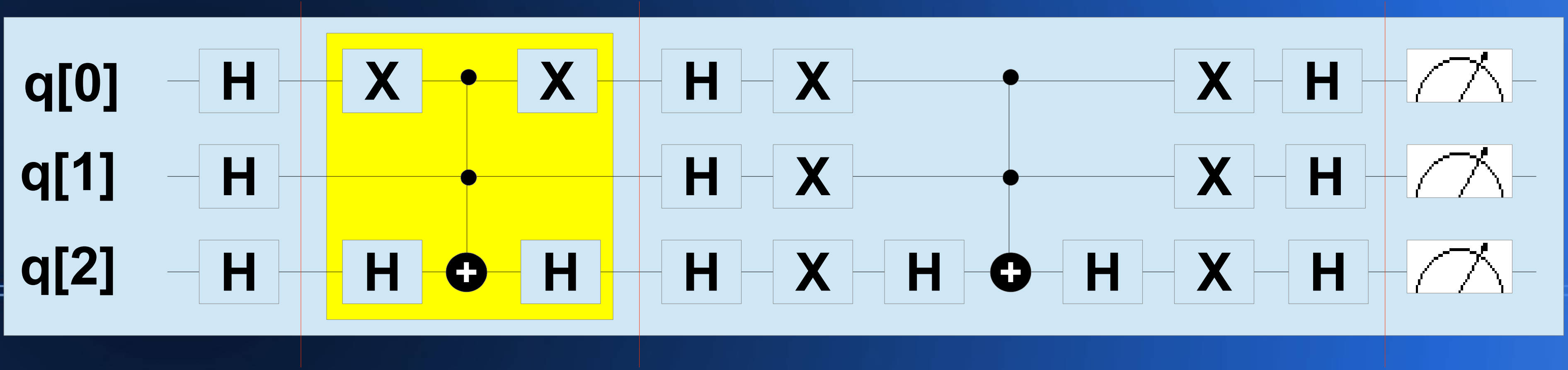
Grover's Algorithm

Requirements (N = # of items to search)

- $\log_2 N$ qubits (e.g. N = 1,000,000 → k = 20)
- An “oracle” function
  - Returns 1 for the element we want
  - Returns 0 for everything else

Grover's Algorithm

1. Place all qubits in superposition (H gates)
2. Run the oracle function on all qubits
3. Use a diffusion operator to increase amplitude of the correct input, decrease the others
4. Measure all qubits



Grover's Algorithm

Result is likely to be correct, but not guaranteed

- Repeat oracle + diffusion
  - Pi * sqrt(N) / 4 = optimal number of iterations
- Or call the oracle function again to verify the result

Shor's Algorithm

- Given a number N, find a nontrivial factor (not 1 or N)
- Uses patterns in modulo arithmetic to make pretty good guesses at the factors

Shor's Algorithm

Consider the sequence:

$x \bmod N$, $x^2 \bmod N$, $x^3 \bmod N$, etc.

Powers of 2	2	4	8	16	32	64	128	256	512
Mod 15	2	4	8	1	2	4	8	1	2

Period = 4

Powers of 2	2	4	8	16	32	64	128	256	512
Mod 21	2	4	8	16	11	1	2	4	8

Period = 6

Euler showed that the period evenly divides (p – 1)(q – 1),
as long as x is not divisible by p or q.

Shor's Algorithm

Inverse Quantum Fourier Transform (QFT)

- Transforms a periodic sequence into its period
- Uses interference to amplify the correct answer
- After transformation, measurement indicates the period

Shor's Algorithm

Does it break encryption?

- Requires 4 ($\log_2 N$) + 2 qubits to factor N
  - e.g. 44,743 < 65,536 = 2^{16} requires 66 qubits
- Commercially available: approx. 55 qubits
- Most powerful: 2048 qubits, or N up to 2^{512}

Classes of Problems

- Factoring prime numbers (Shor)
- Searching among elements (Grover)
- Element uniqueness (Andris Ambainis)
- Optimization problems
  - Solving linear systems of equations
  - Machine learning

A good problem has a finite but large number of inputs, and a way to verify the output.

Common Problems

Algorithms are probabilistic

- They reach the correct solution... usually
- Grover's oracle / amplification steps can be optimized
- Run the program multiple times, or confirm the result another way if possible

Common Problems

Decoherence

- On real quantum hardware, qubit states can degrade during program execution
- Causes unexpected results
- Use redundant entangled qubits
- Run the program multiple times, or confirm the result another way if possible

Simulator

- Like a conventional debugger
- Can run backwards
  - Quantum operations are reversible
- Decoherence is never an issue
  - Some simulators can simulate decoherence
- Run as many times as you like

IBM Quantum Computer

- Quantum computer in the cloud
- Free access with IBM account
- Limited qubits per day (15) per user
- Simulator use is unlimited
- Program will run 1024 times

Comparison

Conventional Computer	Quantum Computer
GB of memory	< 1100 qubits
Bit represents 0 or 1	Qubit represents 0, 1, or superposition
Deterministic output	Chance of an incorrect result
Programs run in one direction	Programs can run forward or backward
“Normal” speed	Faster for some classes of problems
Designed for office conditions	May require extreme cold, shielding, etc.
Usual learning curve for new languages	Steep learning curve

Questions?

?

Other Quantum Computing Terms

Quantum Teleportation

- Uses two entangled qubits and two bits to transfer an input qubit state to another location
- Teleports quantum states, not matter
- Requires a way to send the two bits to the target
- Record distance: 89 miles
- Largest to date: Quantum state of an atom

Other Quantum Computing Terms

Quantum Supremacy

- Ability of a quantum computer to do something a conventional computer cannot
- Google article in Nature (Oct 2019): 200 sec vs. 100 million years to check numbers for randomness

Further Reading

- The Quantum Zoo by Marcus Chown (2006)
- The Amazing Story of Quantum Mechanics by James Kakalios (2010)
- Natural Computing by Dennis Shasha and Cathy Lazere (2010)
- Through Two Doors at Once by Anil Ananthaswamy (2018)
- Quantum Computing for Babies by Chris Ferrie (2018)

On The Web

- http://www.quantum-inspire.com
- http://www.quantumplayground.net/#/home
- https://quantum-computing.ibm.com/login
- YouTube: PBS Quantum Mechanics
- https://ai.googleblog.com/2019/10/quantum-supremacy-using-programmable.html
- https://www.nature.com/articles/nature11472

On The Web

- https://quantumcomputingreport.com/scorecards/qubit-count/
- http://www.alienryderflex.com/polarizer/
- https://aws.amazon.com/braket/

Thank You!