

# Pengantar

Hendrik Santoso Sugiarto

IBDA4221 – Selected Topic in Computer Technology

*Quantum Computing*

# Ekspektasi

- Apakah ekspektasi anda terhadap kelas ini?

# Quantum Computing



# Capaian Pembelajaran

- Silabus
- Qiskit
- Teknologi Kuantum
- Komputer Kuantum
  - Perkembangan Komputer Kuantum
  - Keunggulan Komputer Kuantum

# Silabus

# Logistik

- Jam kuliah : Selasa (14 – 17 WIB, 3x50 menit)
- Tempat : Lab Komputer
- Harap membawa laptop ketika menghadiri kelas
- Jika berhalangan menghadiri sesi tersinkronkan, hubungi dosen lewat e-mail dan SAS di 081-1322-2301
- *Office hours* : by appointment
- E-mail : [Hendrik.sugiarto@calvin.ac.id](mailto:Hendrik.sugiarto@calvin.ac.id)  
Dibalas paling lambat 48 jam (hari kerja) / 72 jam (*weekends*, hari libur)
- Komunikasi : Canvas (silabus, materi kuliah, tugas) maupun e-mail

# Pustaka

- Qiskit Textbook <https://qiskit.org/textbook-beta> (IBM Quantum)
- Isaac Chuang and Michael Nielsen, Quantum Computation and Quantum Information, Cambridge University Press, 2000
- J.J.Sakurai and Jim Napolitano, Modern Quantum Mechanics, Pearson, 2010

# Rincian Nilai Perkuliahan

## **Tugas individu (30%): 5 points x 6 tugas wajib (+ tugas bonus)**

- Tugas individu dikerjakan secara pribadi dan diberikan melalui Canvas untuk menguji pemahaman mahasiswa terhadap materi yang diberikan.
- Tugas dapat berbentuk teori singkat maupun pemrograman (coding) dengan Python dan soal-soal dengan jawaban pendek.
- Setelah melalui batas waktu, nilai maksimal tugas tersebut menjadi 50% sampai 24 jam setelah batas waktu.
- Jika melalui lebih dari 24 jam setelah batas waktu, tugas tersebut tidak mendapat nilai (atau mendapat nilai nol).
- Diperbolehkan mengumpulkan tugas lebih dari sekali (unlimited attempts) melalui Canvas, namun dosen hanya menganggap versi paling terakhir sebagai versi yang mahasiswa secara resmi kumpulkan, dan hanya versi tersebut yang dikoreksi.
- Harap menghubungi dosen jika berhalangan untuk mengumpulkan tugas sebelum deadline karena alasan darurat.



# Rincian Nilai Perkuliahan

- **Ujian Tengah Semester (UTS) (30%)**

Ujian Akhir Semester akan diadakan pada minggu terakhir dalam bentuk pilihan ganda (multiple choice questions) dan esai melalui Canvas.

- **Proyek Ujian Akhir Semester (UAS) (40%)**

Ujian Akhir Semester akan diadakan pada minggu terakhir dalam bentuk proyek

# Tips untuk berhasil dalam MK ini

- Konsisten mengerjakan tugas dan proyek sebaik mungkin. Dosen mengoreksi dengan mendetil sehingga menjadi *feedback* bagi mahasiswa.
- Bertanya dan partisipasi secara aktif di dalam kelas, sehingga dosen juga mendapat *feedback* jika terlalu cepat atau ada yang kurang jelas.
- Tidak tertinggal pelajaran: jika tidak paham di awal, segera menghubungi dosen atau bertanya kepada mahasiswa lainnya.
- Gunakan *office hours* dosen yang sudah dijadwalkan, untuk lebih memahami materi dan tugas.
- Mata kuliah ini banyak memakai pemrograman dengan Python dengan *Qiskit*. Bacalah dokumentasi dari *Qiskit* untuk lebih terbiasa menggunakannya.

# Setelah Anda menyelesaikan perkuliahan ini, Anda akan:

- Memahami dasar matematis/teoretis yang digunakan di dalam *Quantum Computing*
- Mampu membuat algoritma *Quantum* untuk menyelesaikan berbagai macam permasalahan

# Qiskit

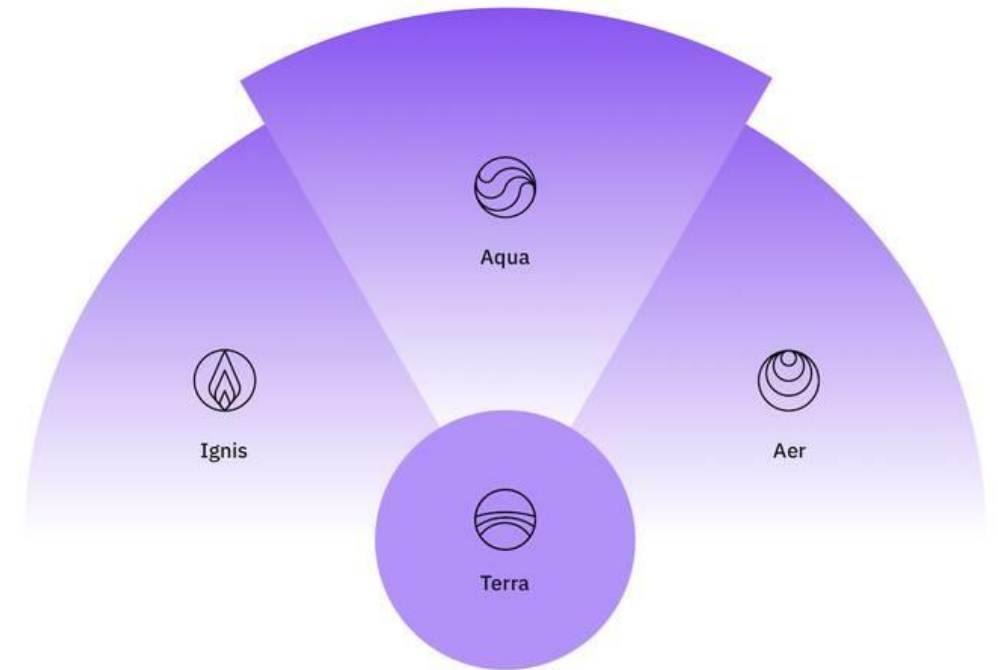
# Anaconda

- Anaconda adalah sistem manajemen paket python
- Melalui Anaconda, maka versi dari paket python yang ada akan dikelola oleh package management system conda.
- Dapat di download untuk windows, linux, dan mac (silahkan mengikuti arahan di website anaconda)
- Untuk custom environment conda silahkan ikuti arahan berikut:  
<https://docs.anaconda.com/ae-notebooks/user-guide/adv-tasks/work-with-environments/>
  - Misal buat environment baru khusus kelas ini: conda create -n quantum
  - Untuk mengaktifkan environment ini: source activate quantum
  - Untuk mematikan environment ini: source deactivate



# Install Qiskit

- <https://learn.qiskit.org/course/ch-prerequisites/environment-setup-guide-to-work-with-qiskit-textbook>
- `pip install qiskit`
- `pip install qiskit[visualization]` → tambahan visualisasi
- Di dalam qiskit terdapat Terra, Aer, Ignis, Aqua:
  - Terra: circuit, interface
  - Aer: simulator
  - Ignis: error correction
  - Aqua: domain applications
- `import qiskit` → menggunakan qiskit
- `qiskit.__qiskit_version__` → cek versi

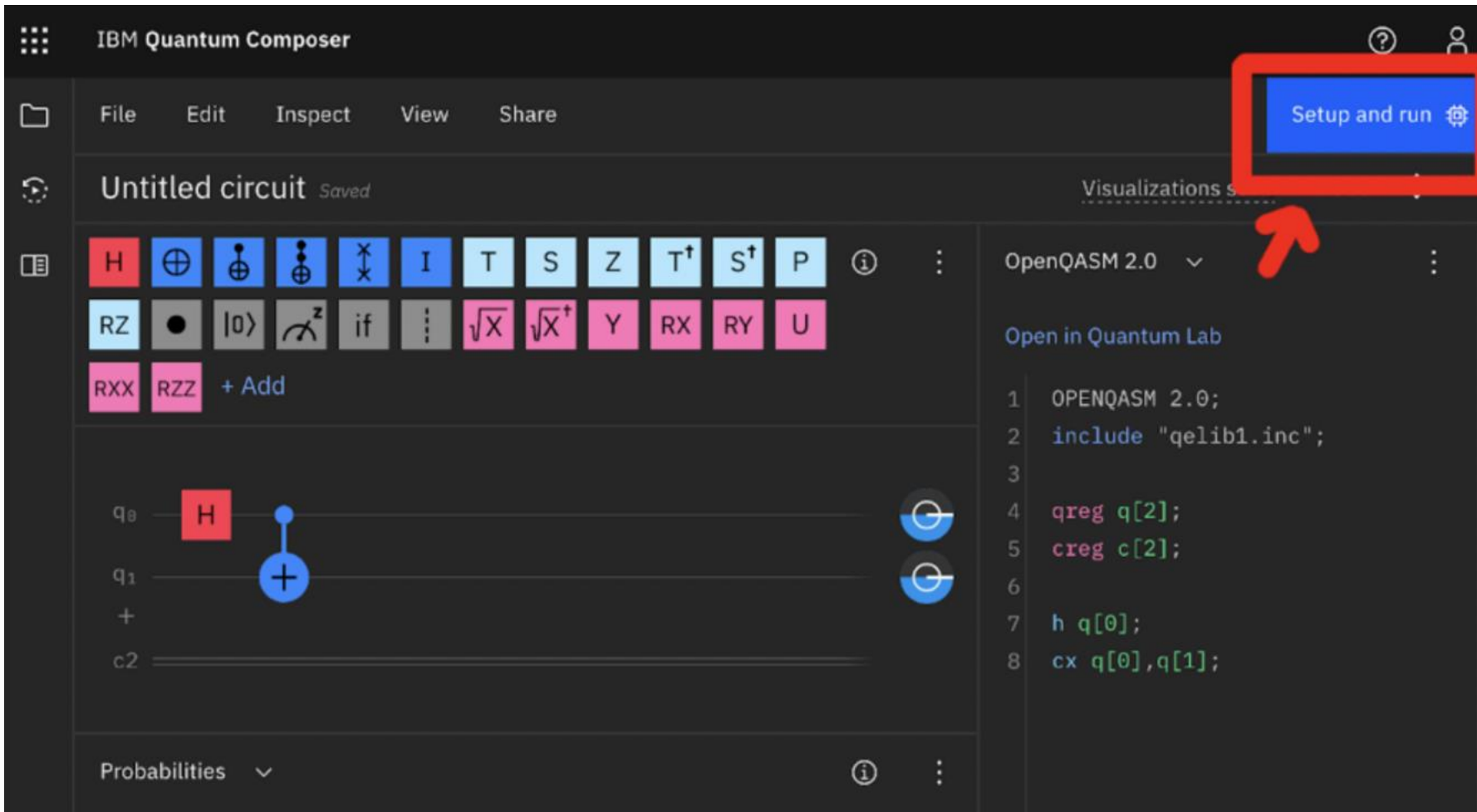


# Akses IBM Quantum

- Masuk ke <https://quantum-computing.ibm.com/>
- Gunakan email calvin
- 2 cara menggunakan computer kuantum:
  - Melalui composer
  - Melalui Qiskit

# Membuat Sirkuit Kuantum menggunakan Composer

- Setelah merangkai sirkuit klik setup and run





# Memilih Quantum Cloud

- Ganti provider menjadi “ibm-q-education”

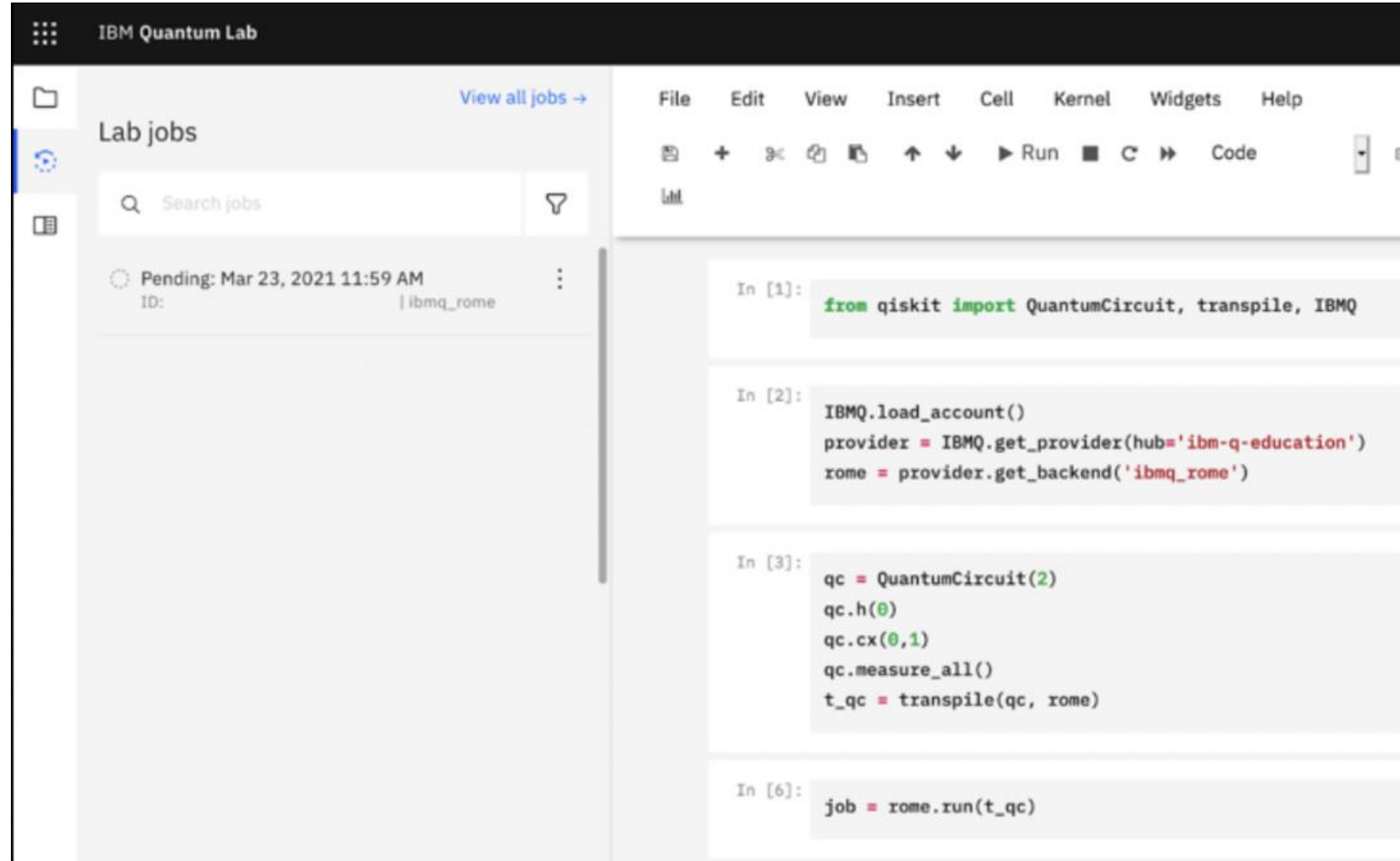
The screenshot displays the 'Set up and run your circuit' interface, divided into two main steps:

- Step 1: Choose a system or simulator**
  - A search bar labeled 'Search by system or simulator name' is at the top.
  - Below the search bar, there are three placeholder cards, each with a radio button and a 'See details' link. A red arrow points to the bottom card with the text 'Select backend'.
  - At the bottom, a card for 'ibmq\_5\_yorktown' is selected (indicated by a filled radio button). It shows 'System status' and 'Total pending jobs' as '5 Qubits' and '8 Quantum volume' respectively. A red arrow points to this card with the text 'Select backend'.
- Step 2: Choose your settings**
  - The 'Provider' dropdown menu is set to 'ibm-q/open/main'. A red arrow points to this dropdown with the text 'Select provider'.
  - The 'Shots' dropdown menu is set to '1024'.
  - The 'Job limit' is '5 remaining'.
  - Under 'Optional', the 'Name your job' section has a text input field with the placeholder 'e.g. Untitled circuit job'.
  - The 'Tags' section has a text input field with the placeholder 'Add tags'.
  - At the bottom right, a blue button labeled 'Run on ibmqx2' is visible. A red arrow points to this button with the text 'Run Job'.

At the bottom left of the interface is a 'Close' button.

# Membuat Sirkuit Kuantum menggunakan Qiskit

- Load akun dan gunakan backend 'ibm-q-education' provider
- Buat jobs, lalu run pada backend.
- Jika menggunakan IBM Quantum Lab, job akan terlihat pada panel kiri setelah beberapa detik.



The screenshot displays the IBM Quantum Lab web interface. On the left, the 'Lab jobs' panel shows a job with the status 'Pending' at 'Mar 23, 2021 11:59 AM' on the 'ibmq\_rome' backend. The main area on the right contains a code editor with the following Qiskit code:

```
In [1]: from qiskit import QuantumCircuit, transpile, IBMQ

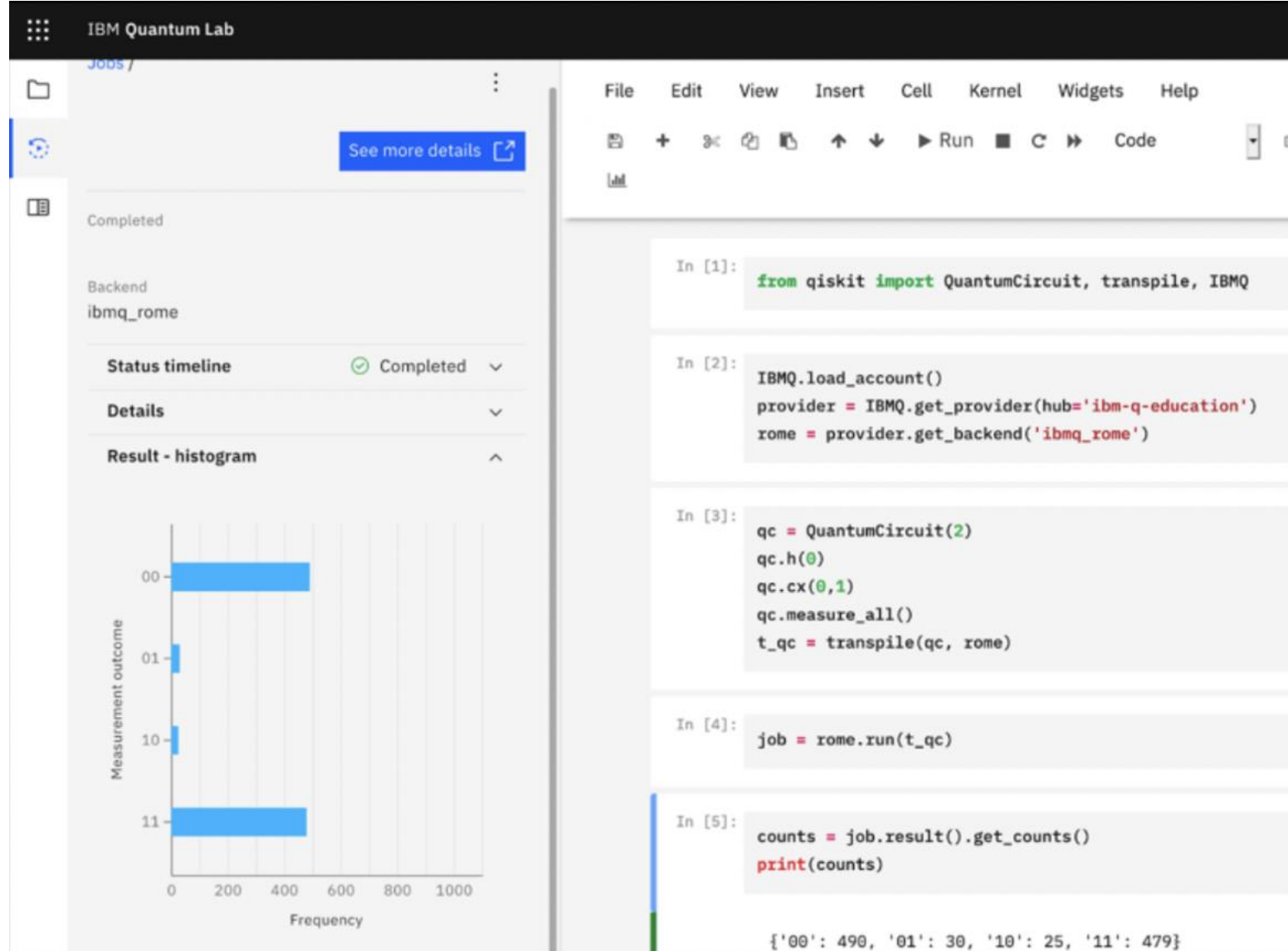
In [2]: IBMQ.load_account()
provider = IBMQ.get_provider(hub='ibm-q-education')
rome = provider.get_backend('ibmq_rome')

In [3]: qc = QuantumCircuit(2)
qc.h(0)
qc.cx(0,1)
qc.measure_all()
t_qc = transpile(qc, rome)

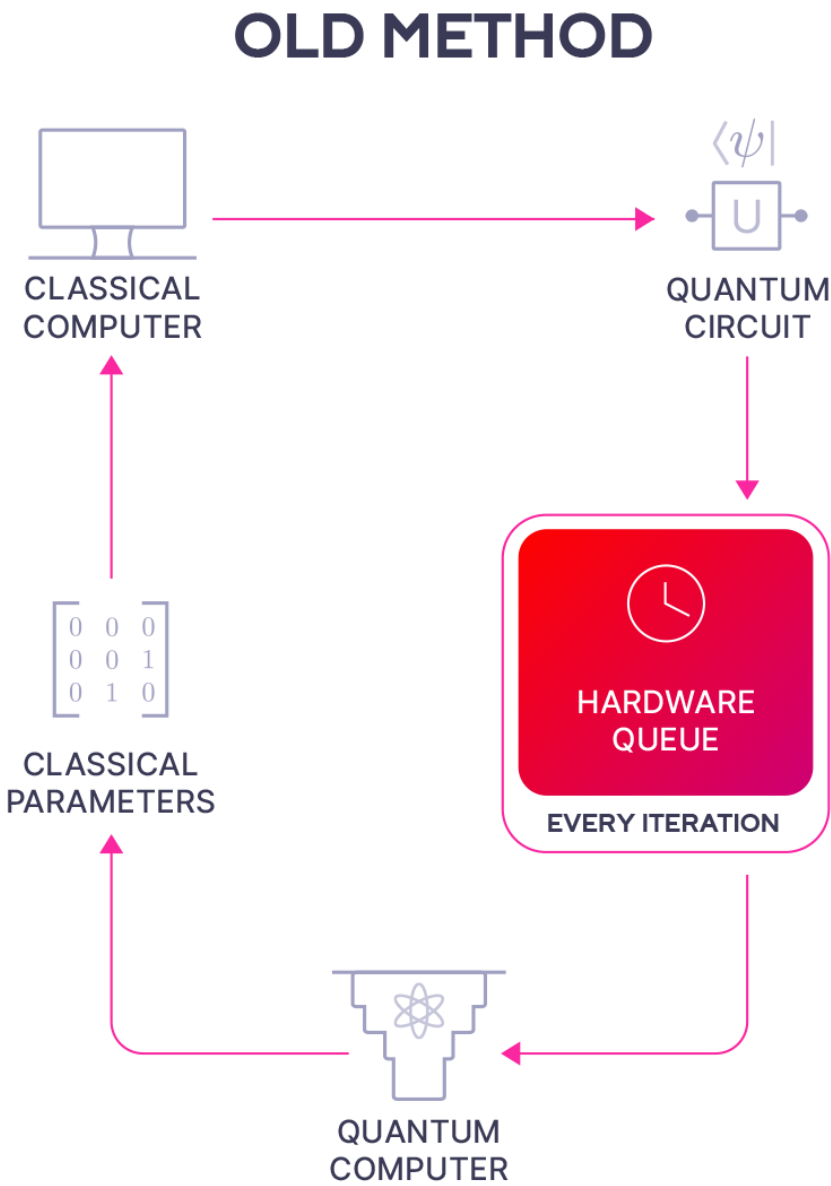
In [6]: job = rome.run(t_qc)
```

# Membuat Sirkuit Kuantum menggunakan Qiskit

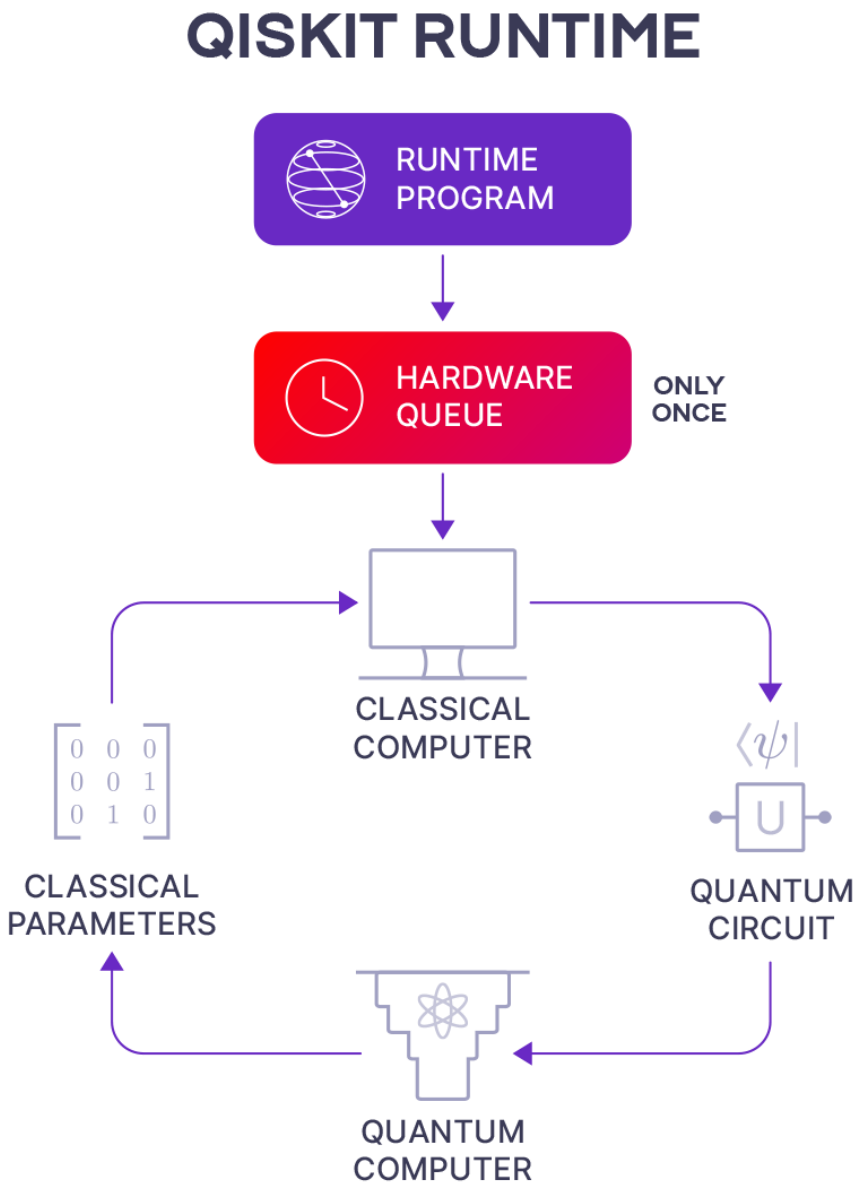
- Hasil akan terlihat pada panel kiri jika job yang dirun sudah selesai



# Qiskit Runtime



VS



# Teknologi Kuantum

# Teknologi Kuantum

- Teknologi kuantum: semua jenis teknologi yang memanfaatkan mekanisme kuantum
- Apakah hanya komputer kuantum?
- Quantum 1.0 vs Quantum 2.0

# Quantum 1.0

- Teknologi yang memerlukan mekanisme kuantum agar dapat berfungsi
- Contoh:
  - Transistors & semiconductor
  - LED
  - Photodetectors
  - Laser
  - Superconductor
  - GPS
  - MRI
  - Solar panel
  - Electron microscope

# Quantum 2.0

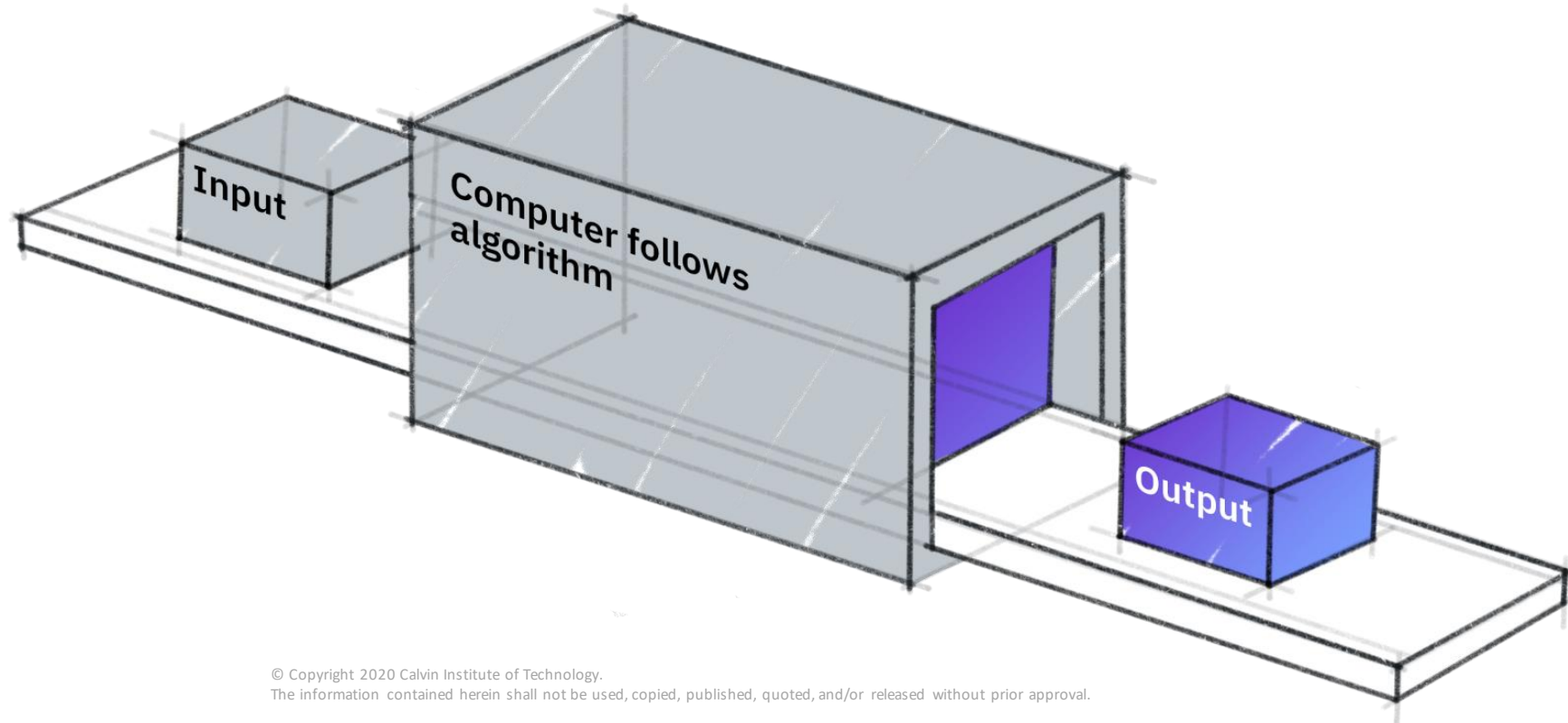
- Teknologi yang membuat dan memanipulasi quantum states dari materi menggunakan superposisi dan entanglement
- Contoh:
  - Quantum information
  - Quantum computing
  - Quantum metrology
  - Quantum simulation



# Perkembangan Komputer Kuantum

# Back to Basic

- Apa itu komputer?
- Komputer mempunyai beragam bentuk: cloud, laptop, smartpone, sistem logistik, dll
- Semuanya memiliki kesamaan: proses pengolahan dari input informasi menjadi output informasi dengan menggunakan himpunan instruksi (algoritma).



# Klasik vs Kuantum

- Klasik

State  $(x, p) \rightarrow$  dynamics  $(F = \frac{dp}{dt}) \rightarrow$  measurement  $(\Delta x, \Delta p) \rightarrow$  observables  $(x, p)$   
Input  $\rightarrow$  algorithm  $\rightarrow$  classical noise + output : classical computing

- Kuantum

$\leftarrow\leftarrow\leftarrow$  quantum bit | classical bit  $\rightarrow\rightarrow\rightarrow$

State  $(\psi) \rightarrow$  dynamics  $(i\hbar \frac{\partial \psi}{\partial t} = H\psi) \rightarrow$  measurement  $(\psi_i) \rightarrow$  observables  $(\langle x \rangle)$   
Preparation  $\rightarrow$  transmission  $\rightarrow$  detection  $\rightarrow$  interpretation : quantum communication  
Initialization  $\rightarrow$  process  $\rightarrow$  measurement  $\rightarrow$  interpretation : quantum computing

# Kompleksitas Algoritma

Kompleksitas:

- Konstan

Linear

Contoh:

- Klasifikasi ganjil/genap

Penjumlahan

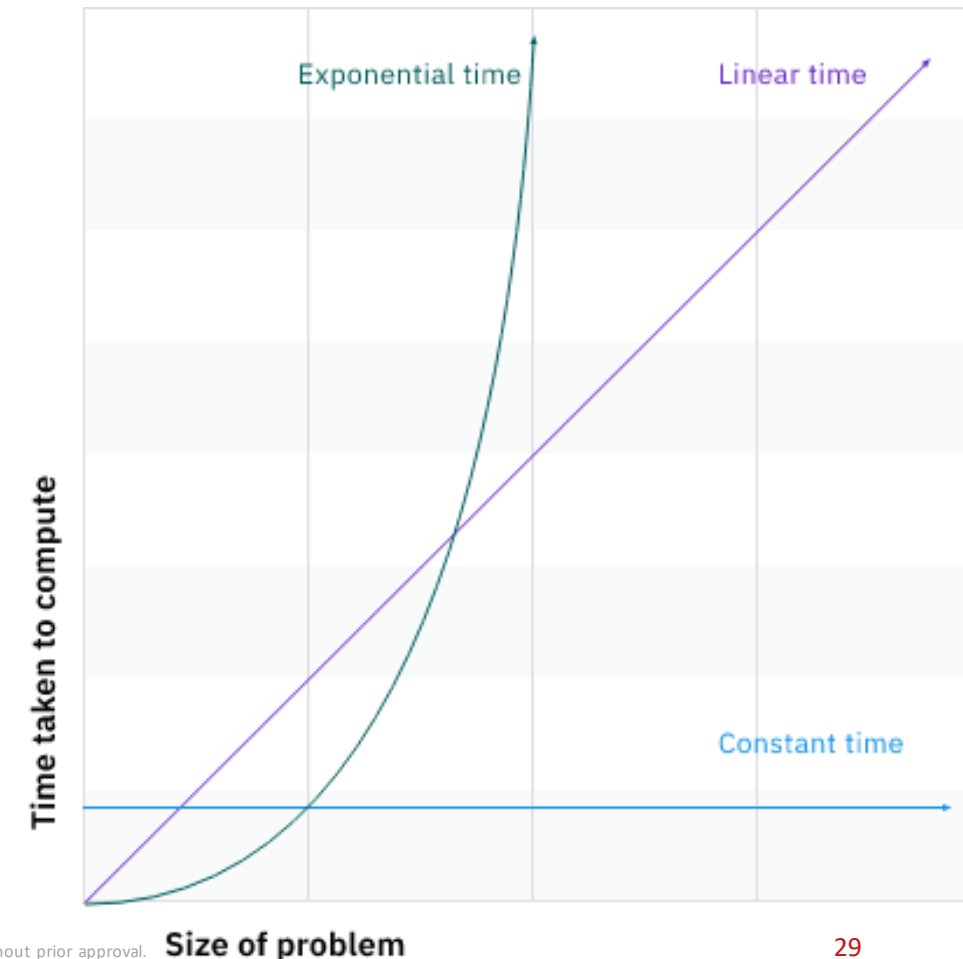
Input	Computer follows algorithm	Output	Input	Computer follows algorithm	Output
13	<pre>let x be the rightmost digit if x is 0, 2, 4, 6 or 8:     output Even else:     output Odd</pre>	Odd	<pre>3247 +1195</pre>	<pre>for each column in the input:     let 's' be the sum of the digits     in that column     if s &gt; 9:         add 1 to next input column to         the left         add s - 10 to output string     else:         add s to output string</pre>	4442

# Algoritma untuk Menebak PIN

- Misalkan kita mempunyai angka rahasia (PIN) dan menebaknya dengan menggunakan cara brute-force (mencoba semua kemungkinan). Berapa lama waktu yang diperlukan?

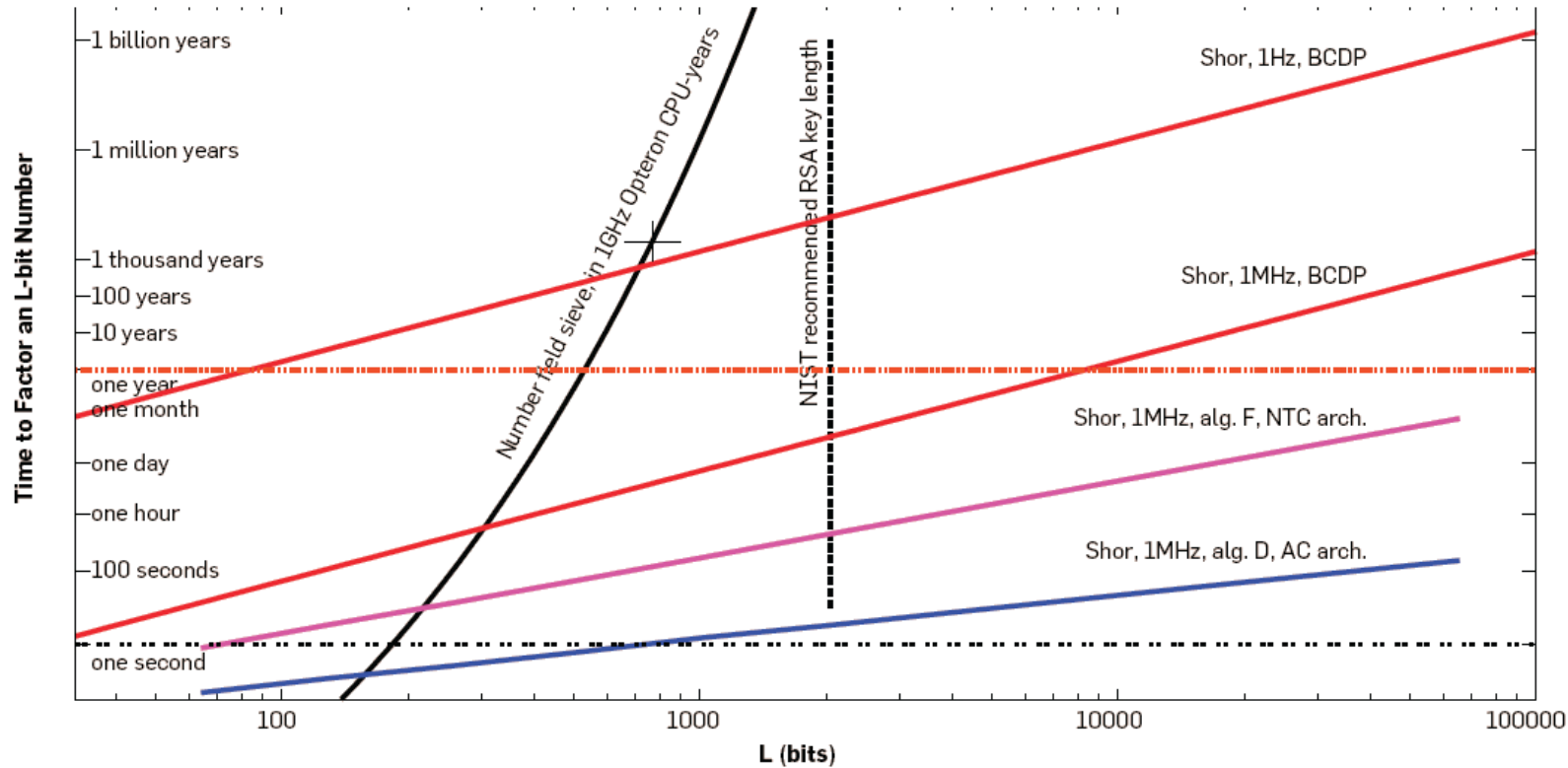
Input	Computer follows algorithm	Output
5672	<pre>choose random x between 0 &amp; 9999 if x matches secret number   output x else:   go to start</pre>	5672

- 1-digit memiliki 10 pilihan (0, 1, 2, 3, 4, 5, 6, 7, 8, 9)
- 2-digit memiliki 100 pilihan ( $T \propto 10^d$ )
- Misalkan 1 detik untuk 10-digit (5 miliar kombinasi/detik)
- Maka kita membutuhkan:
  - ~150 tahun untuk 20-digit PIN
  - ~150 miliar tahun (120x umur alam semesta) untuk 30-digit PIN



# Algoritma Keamanan RSA

- Integer Factorization
- $p \times q = x$
- Contoh:  $13 \times 7 = 91$
- 2700 core year: 250-digit
- Beberapa detik: jutaan qubit

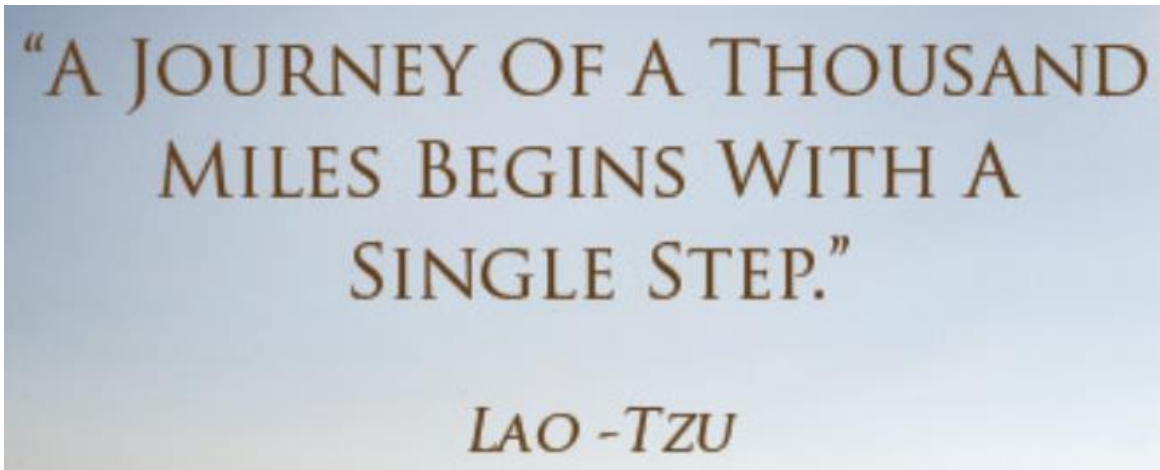


- Current IBM Quantum → 65 qubits
- Future IBM Quantum → >1000 qubits



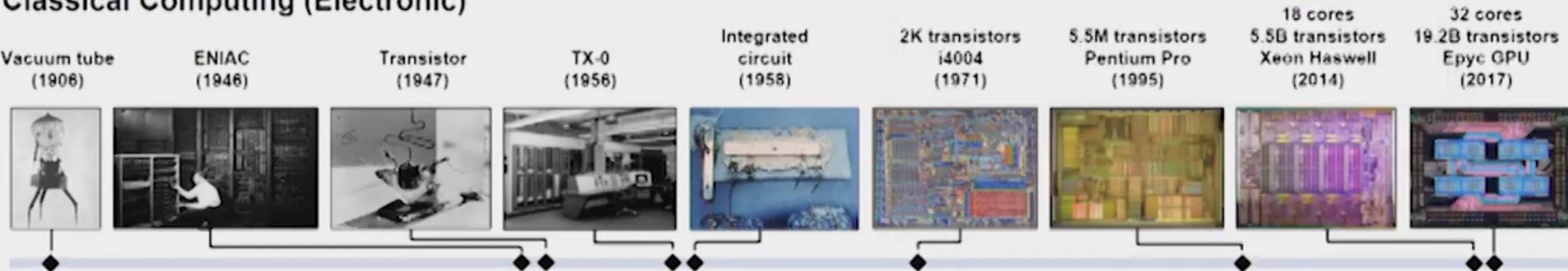
# Small Steps in Technology

- Transistor pertama (1947)
- 70 tahun kemudian:
  - 1 prosesor memiliki miliaran transistor

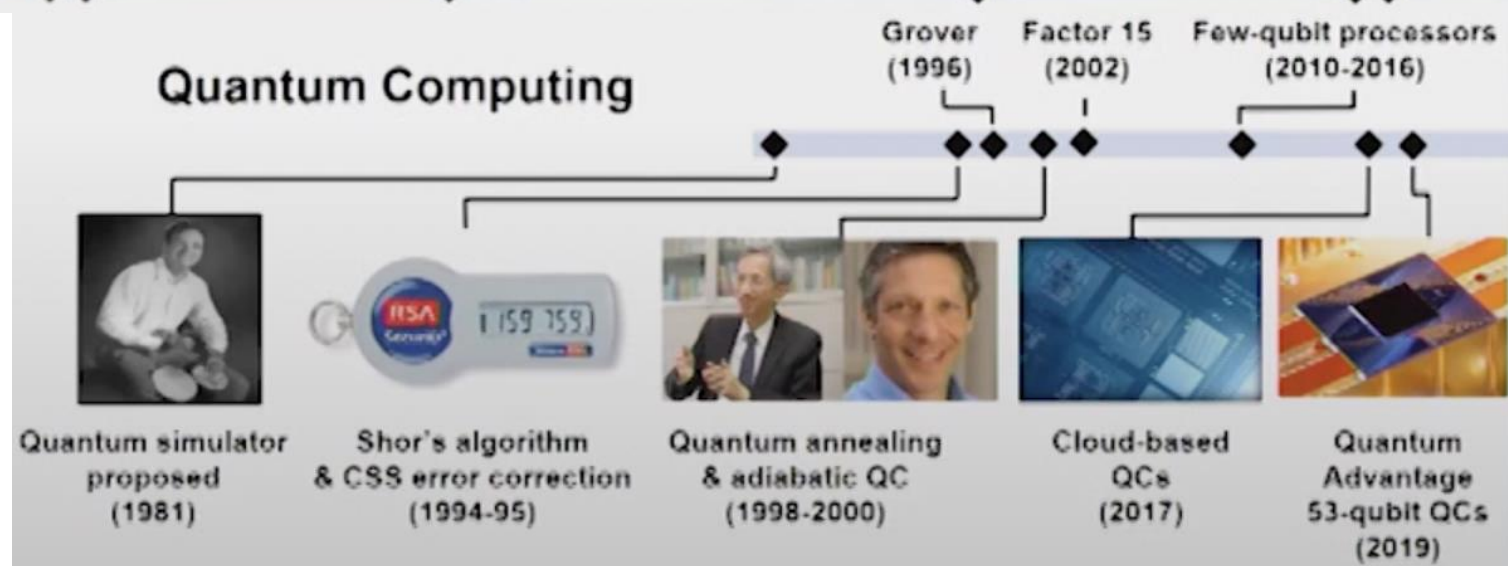


# Timeline

## Classical Computing (Electronic)



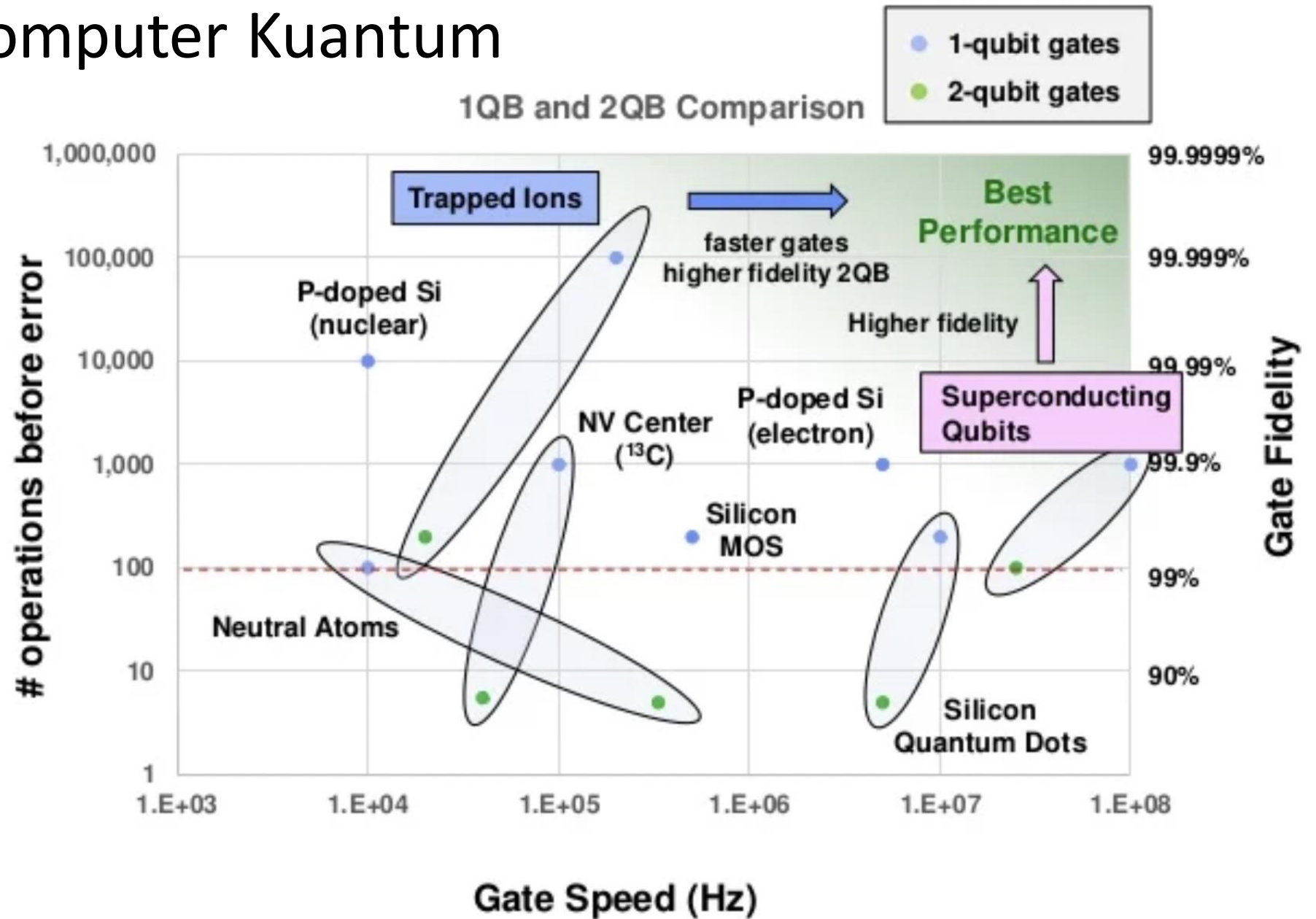
## Quantum Computing



Sama seperti komputer klasik, komputer kuantum masih harus melewati periode development sebelum bisa digunakan secara luas



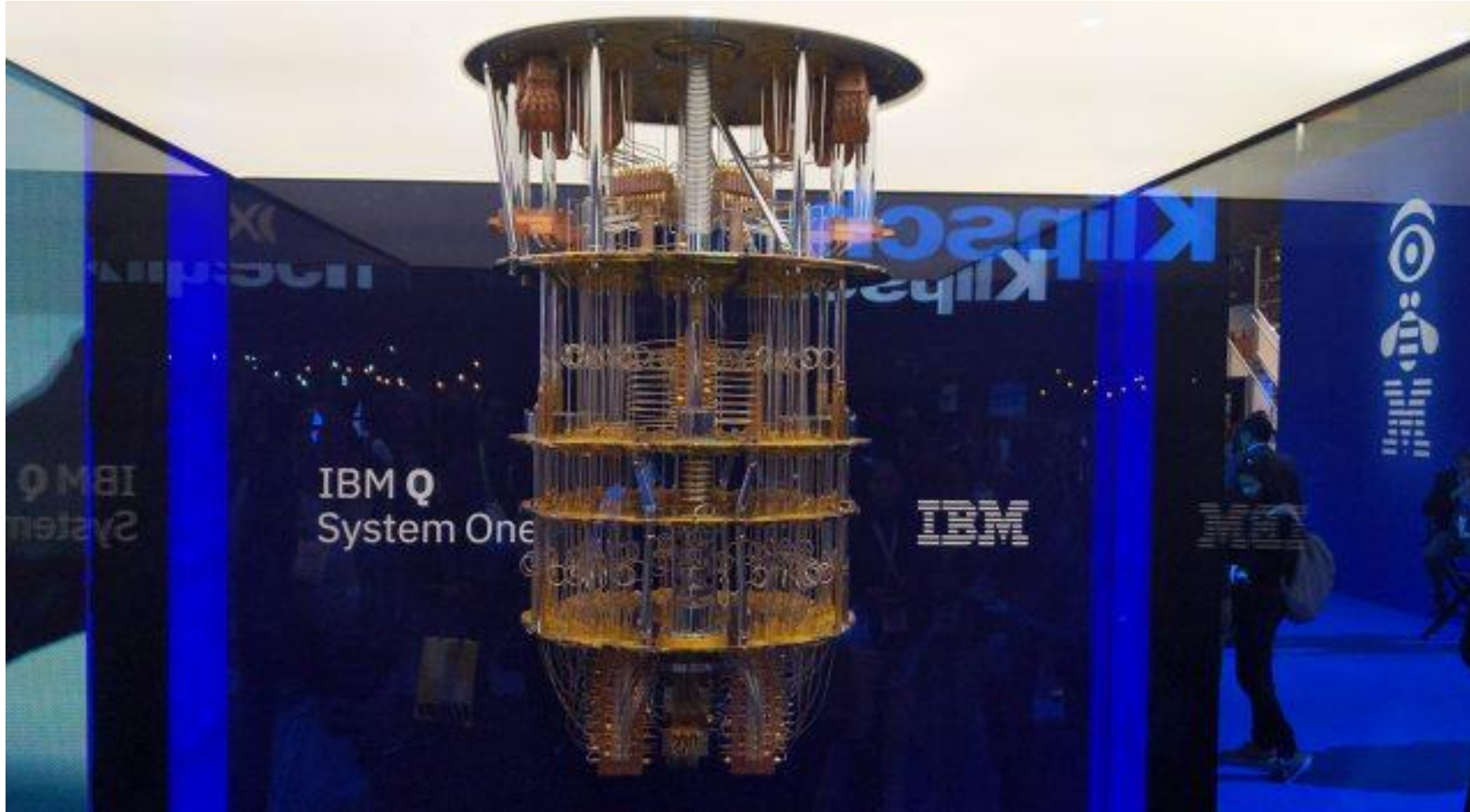
# Kandidat Komputer Kuantum



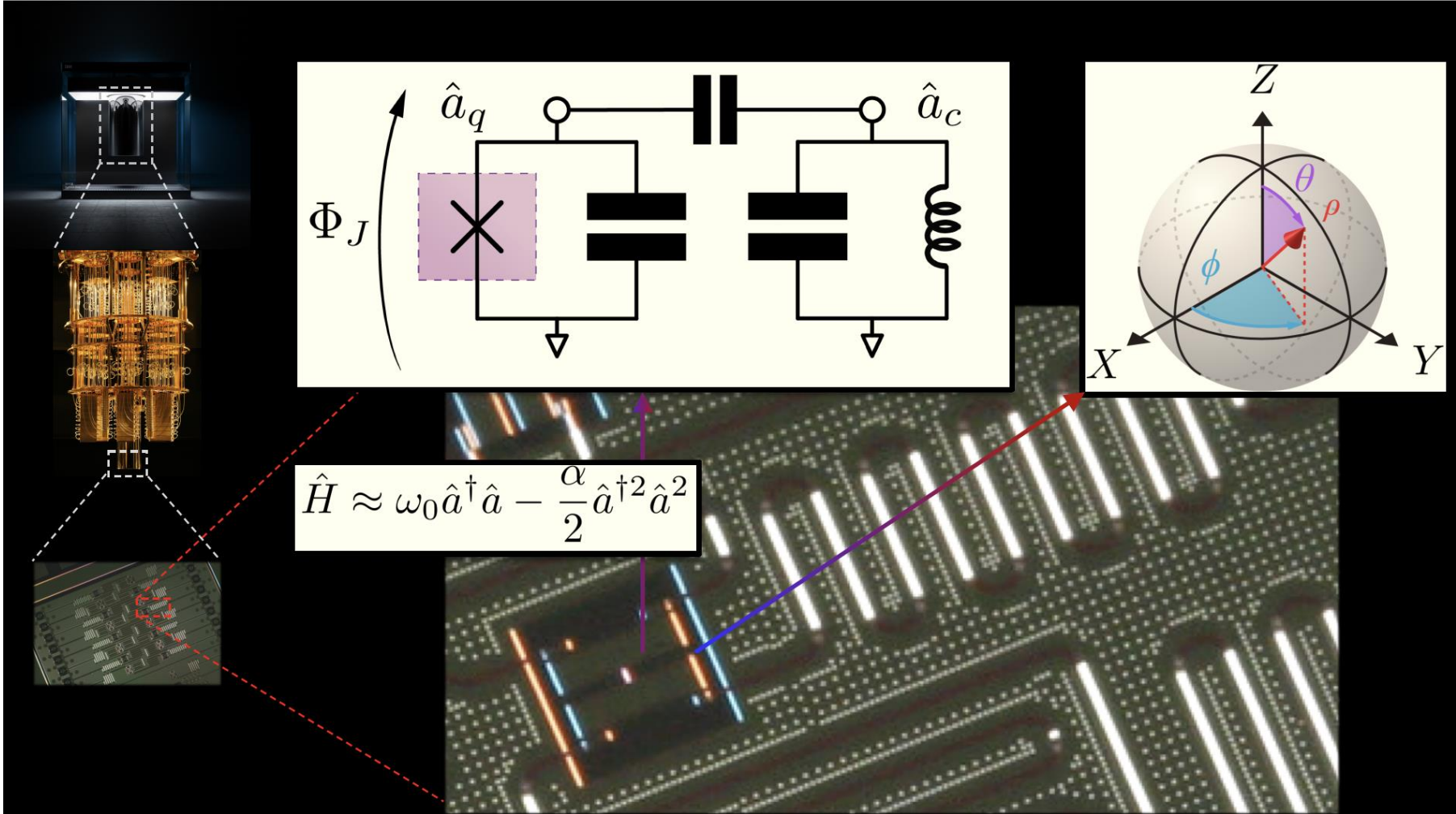
# Perbandingan Paradigma

Paradigma	Kubit	Entanglement	Gate	Skalabilitas	Pemeliharaan	Pengembang
Superconducting	Fluks kubit	Mudah	- Kecepatan Tinggi - Fidelitas Tinggi - Waktu Koherensi	Cukup mudah	- Temperatur 10mK	IBM, Google, rigetti
Photonics	Polaritas foton	Sulit	- Cepat - Fidelitas baik yang menjanjikan - Waktu Koherensi	Sulit	- Temperatur ruang	Xanadu
Trapped Ions	Level energi dari ion yang terperangkap.	Mudah	- Pelan - Fidelitas Tinggi	Agak sulit	- Medan Listrik yang berosilasi (perangkap ion).	IonQ, MIT, Oxford
Neutral Atoms	Energi level atom.	Mudah	- Waktu koherensi yang lama	Sulit	- Vakum temperatur ruang. - Laser sebagai "penjepit" optikal.	Pasqal
Quantum Dots	Putaran Elektron	Cukup mudah	- Fidelitas tinggi	Sulit namun berpotensi semakin mudah	- Kotak Kuantum - Ruang gelap pada suhu 4° Celsius.	Intel
Diamond Vacancies	Level energi elektron	Sulit	- Waktu Koherensi yang baik - Gerbang 2 kubit memiliki fidelitas rendah	Cukup sulit	- Temperatur ruang - vakum Tingkat tinggi.	Brilliance
NMR	Putaran Nuklei	Mudah	- Efisien - Waktu Koherensi yang baik. - fidelitas yang menjanjikan (jumlah kubit kecil)	Sulit	- Temperatur ruang	SpinQ
Topological	Arah anyon	Mudah	- Tahan terhadap gangguan	Cukup mudah	- Medan magnet yang kuat. - Topologi superkonduktor, superkonduktor-isolator topologi/semikonduktor	Microsoft
Bosonic	Superposition of energy levels	Mudah	- Tahan terhadap gangguan	Cukup mudah	- Perlindungan QEC. - Arsitektur QED.	Delft, Tsinghua

# Superconducting Qubits

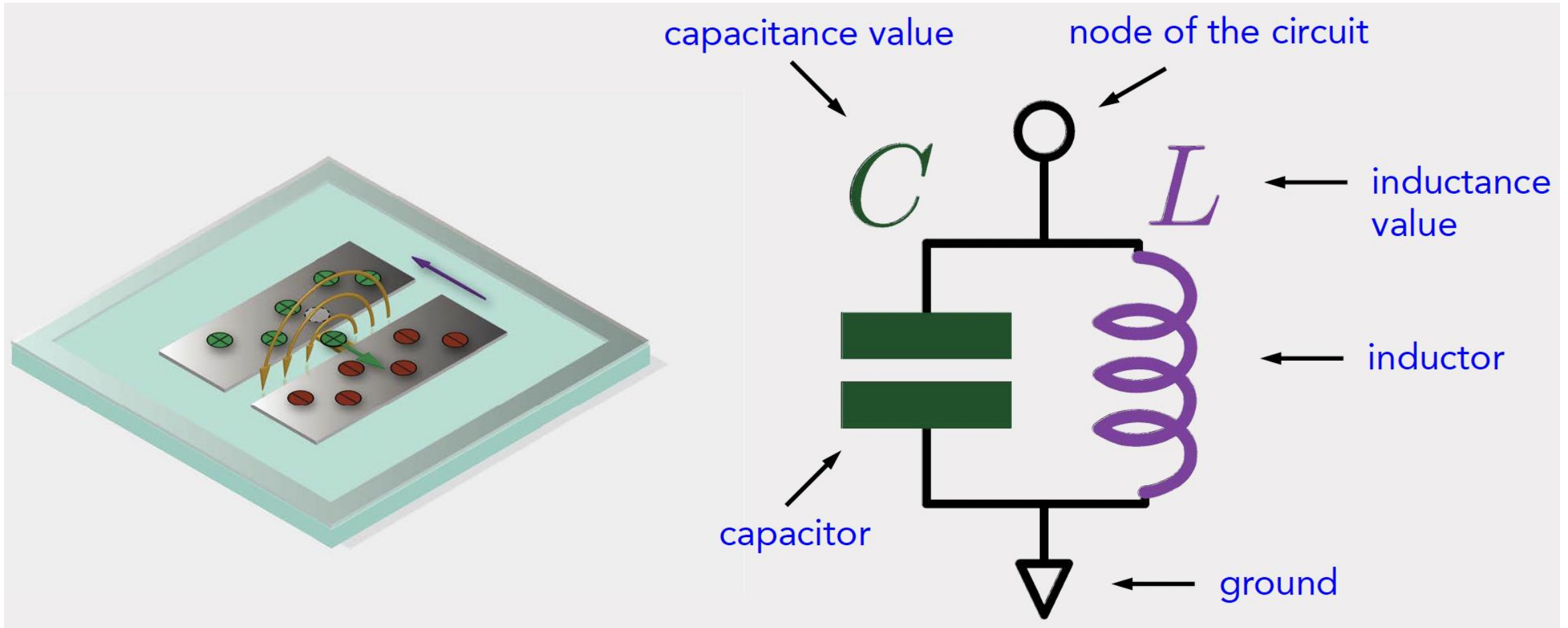


# Quantum Oscillator



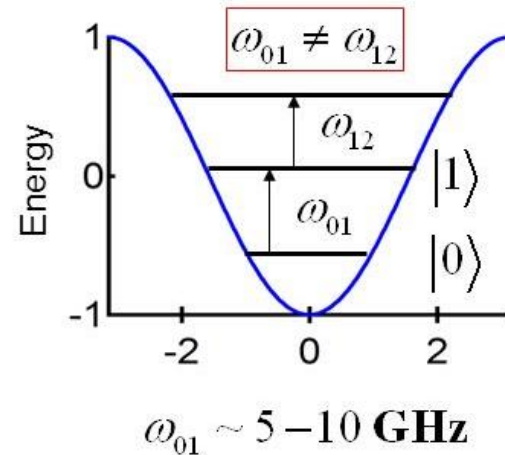


# Transmon Qubit



# Anharmonic Quantum Oscillator

- <https://qiskit.org/textbook/ch-quantum-hardware/transmon-physics.html>



Anharmonicity allows us to approximately treat oscillator as a two-level 'spin'.

$$|1\rangle = |\uparrow\rangle$$

$$|0\rangle = |\downarrow\rangle$$

Examples of coherent superpositions

$$\frac{1}{\sqrt{2}}(|\downarrow\rangle + |\uparrow\rangle) = |\rightarrow\rangle$$

$$\frac{1}{\sqrt{2}}(|\downarrow\rangle - |\uparrow\rangle) = |\leftarrow\rangle$$

# Keunggulan Komputer Kuantum

# Quantum Supremacy

Algorithm	Classical Time	Quantum Time	Speedup	Limitation
<b>Factoring<sup>1</sup></b> (+ related number theoretic)	$2^N$ (for N digits)	$N^3$	Exponential	Classical runtime limit unproven
<b>Simulation<sup>2</sup></b> (quantum chemistry)	$2^N$ (for N atoms)	$N^c$	Exp. in space, polynomial in time	Mapping problem to qubits
<b>Linear systems<sup>3</sup></b> ( $Ax=b$ )	$2^N$ (for N digits)	$\sim N$	Exponential	Strict conditions, e.g. sparse matrix
<b>Optimization<sup>4</sup></b>	$2^N$	?	?	Empirical
<b>Search<sup>5</sup></b> (unsorted / unstructured data)	$N$	$\sqrt{N}$	Polynomial ( $\sqrt{N}$ )	Data loading



# Keunggulan Komputer Kuantum

- Superposisi
- Non-local (spooky action at distant)
- Kapasitas (peningkatan kompleksitas eksponensial):
  - 20 qubits =  $2^{20} > 1$  juta kemungkinan
  - 100 qubits > semua bits pada hard-drive dari seluruh dunia
  - 300 qubits > seluruh partikel dalam alam semesta
- Performa (bahkan sanggup menembus enkripsi terkuat di dunia)
- Efisiensi (kalor yang dikeluarkan untuk tiap komputasi hampir nol)

# Aplikasi Komputer Kuantum

- Memodelkan interaksi kuantum (drug discovery, protein folding, chemical bonding, solid state properties, neuroscience)
- Kriptografi (enkripsi, keamanan)
- Simulasi (finance, logistic, forecasting)
- Optimisasi (machine learning, matrix factorization, neural network)

# Summary

# Tuhan Memberkati