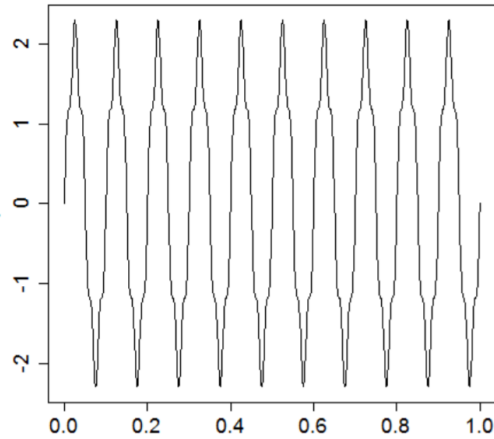
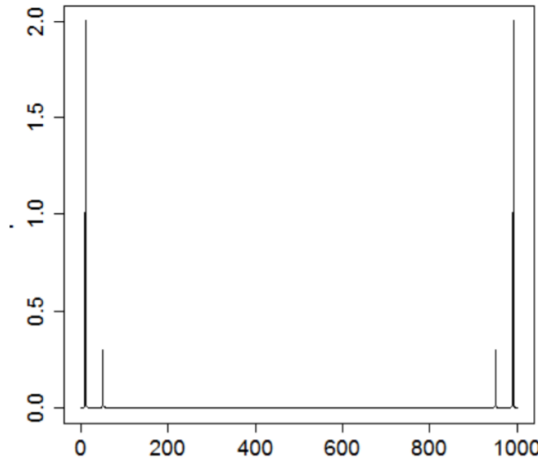
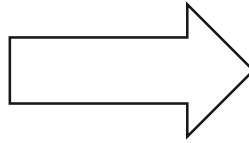


# The Quantum Fourier Transform and Its Applications

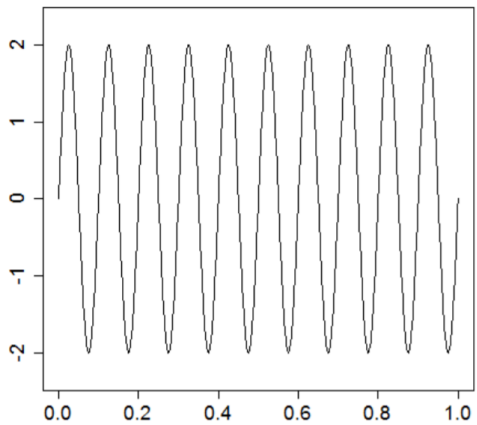
# Discrete Fourier Transform



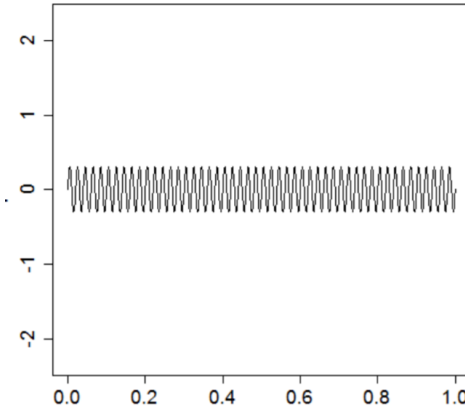
Input signal



Input signal in frequency domain



True signal:  $f=10$ ,  $A=2$

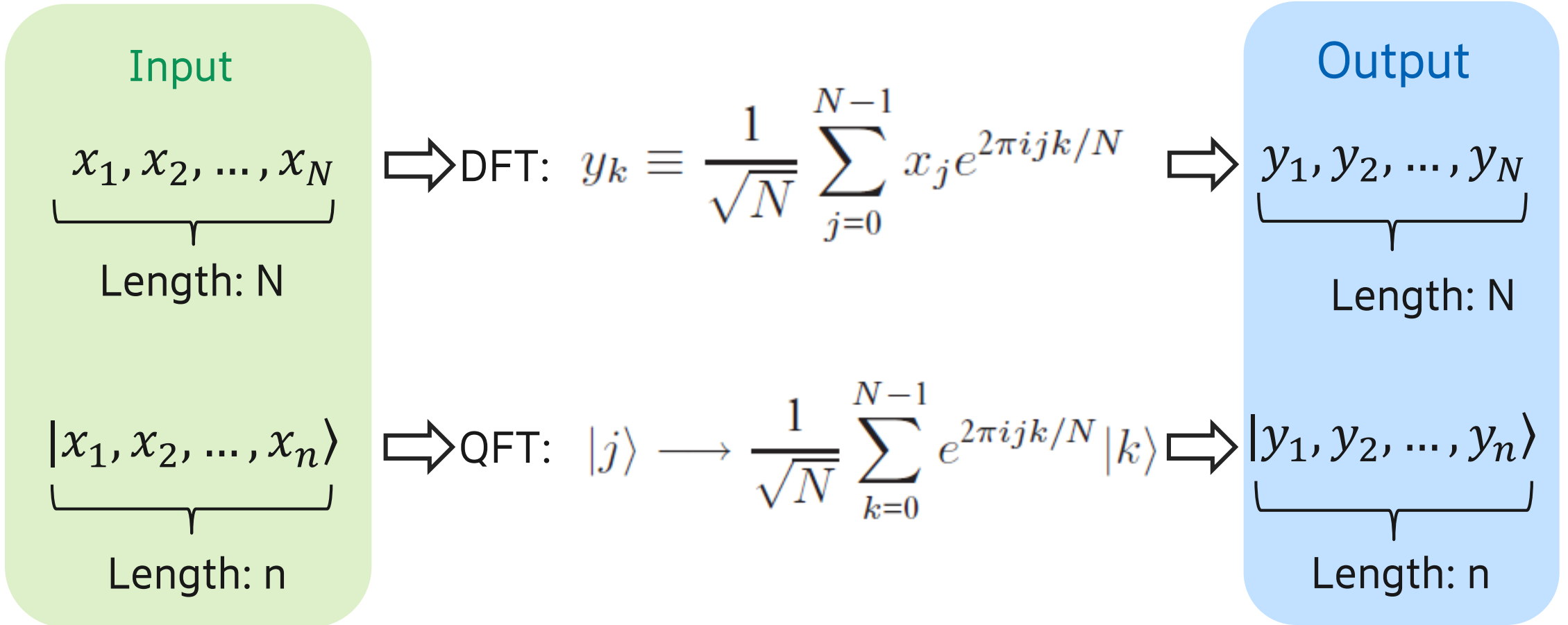


Noise signal:  $f=50$ ,  $A=0.3$

QFT is the quantum analogue of DFT:

- Map signal from **time** domain to **frequency** domain on quantum computer with a faster speed
- Basis for **quantum phase estimation**

# Quantum Fourier Transform



$$N = 2^n$$

# Quantum Fourier Transform

$$\text{QFT: } |j\rangle \longrightarrow \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2\pi i j k / N} |k\rangle$$

$$\underbrace{|x_1, x_2, \dots, x_n\rangle}_{\text{Length: } n} \Rightarrow \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & 1 & 1 & 1 & \dots & 1 \\ 1 & \omega_n & \omega_n^2 & \omega_n^3 & \dots & \omega_n^{N-1} \\ 1 & \omega_n^2 & \omega_n^4 & \omega_n^6 & \dots & \omega_n^{2(N-1)} \\ 1 & \omega_n^3 & \omega_n^6 & \omega_n^9 & \dots & \omega_n^{3(N-1)} \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ 1 & \omega_n^{N-1} & \omega_n^{2(N-1)} & \omega_n^{3(N-1)} & \dots & \omega_n^{(N-1)(N-1)} \end{bmatrix} \Rightarrow \underbrace{|y_1, y_2, \dots, y_n\rangle}_{\text{Length: } n}$$

N×N matrix

where  $\omega_n := e^{\frac{2\pi i}{2^n}}$

Example (2 qubits):

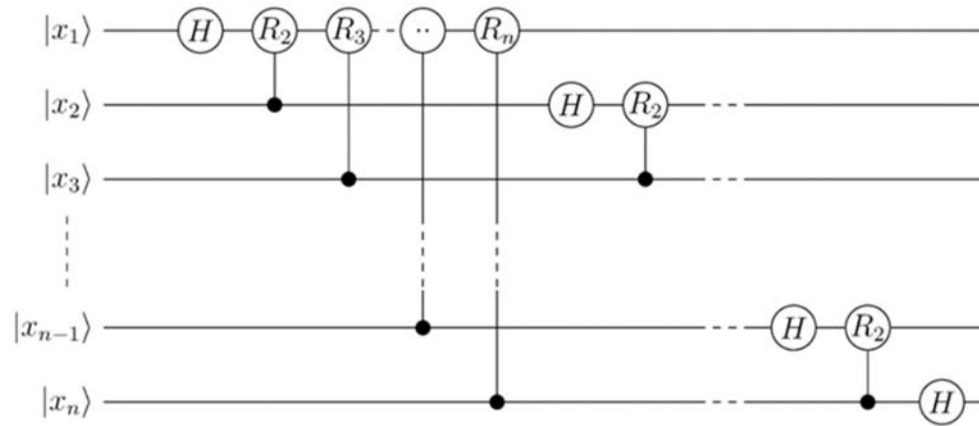
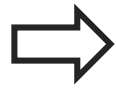
$$F_4 = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{bmatrix}$$

# Quantum Fourier Transform

$$\text{QFT: } |j\rangle \longrightarrow \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2\pi i j k / N} |k\rangle$$

$|x_1, x_2, \dots, x_n\rangle$

Length: n



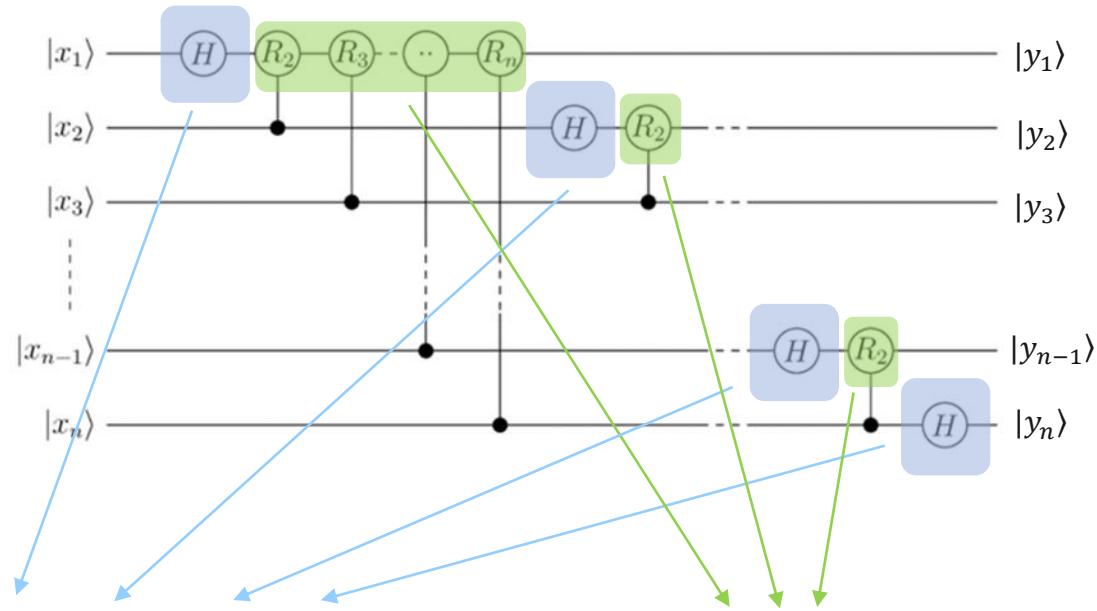
Quantum circuit for QFT

$|y_1, y_2, \dots, y_n\rangle$

Length: n

# Quantum Fourier Transform

Quantum circuit for QFT

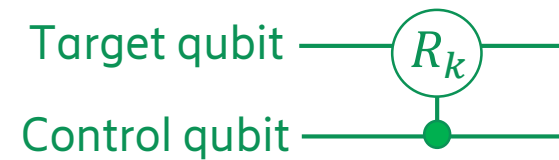


Hadamard gate: create superposition

$$|0\rangle \rightarrow \frac{|0\rangle + |1\rangle}{\sqrt{2}}$$

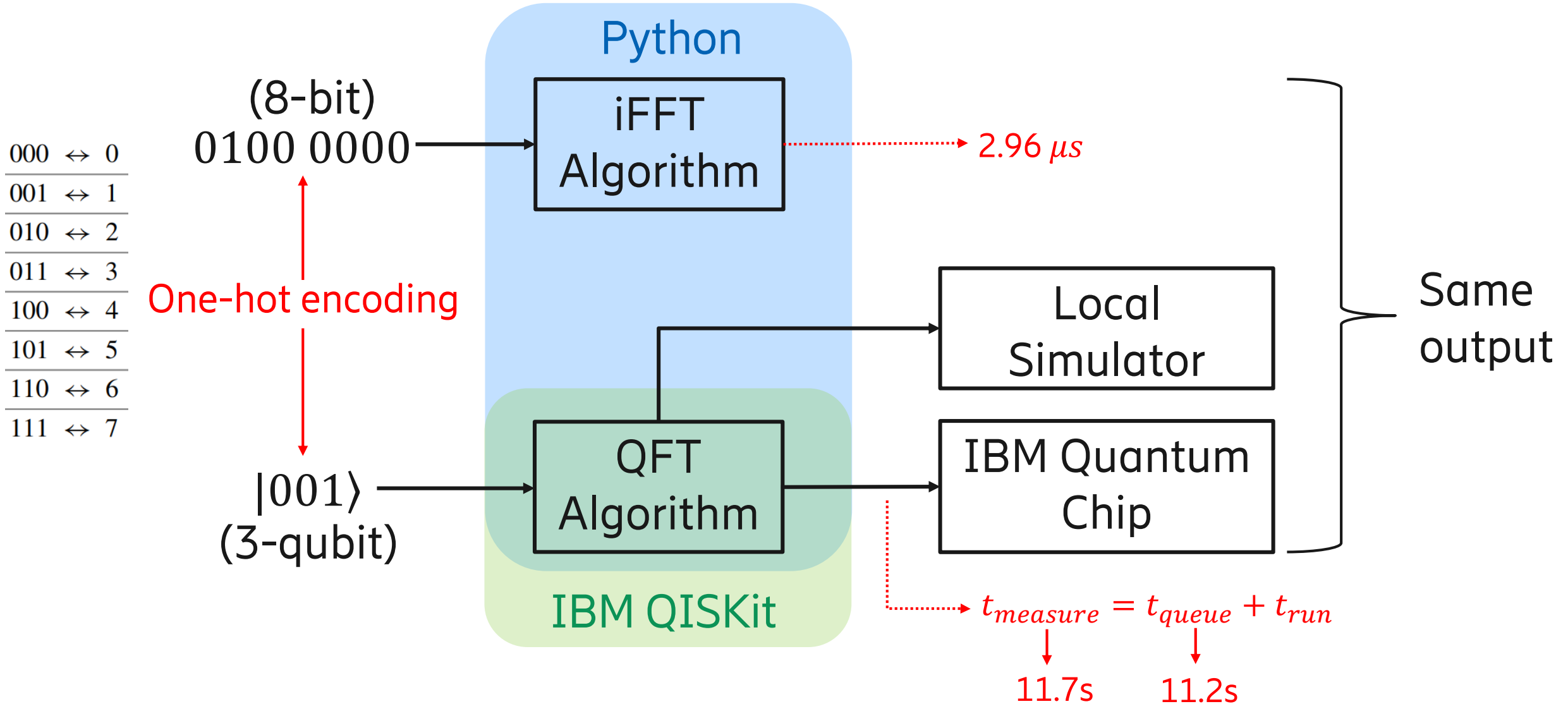
$$|1\rangle \rightarrow \frac{|0\rangle - |1\rangle}{\sqrt{2}}$$

Controlled phase gate



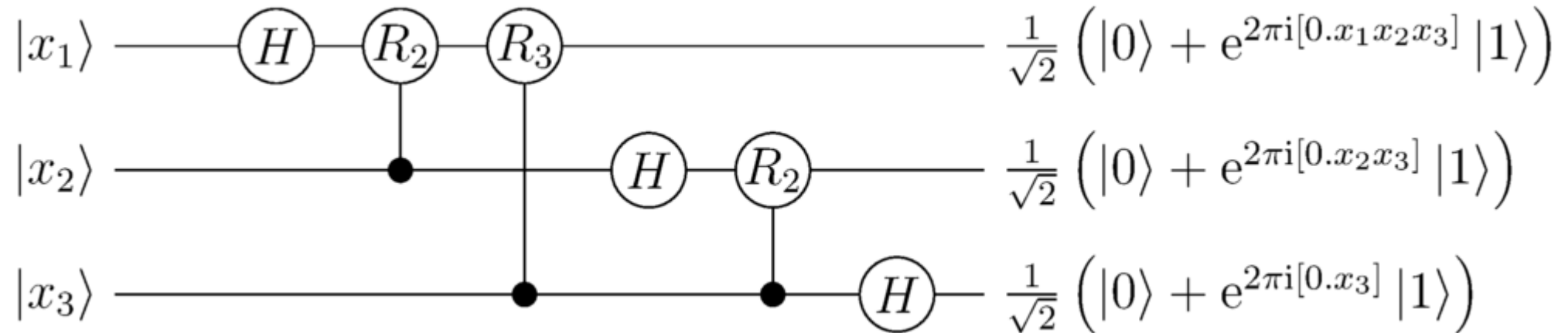
$$R_k \equiv \begin{bmatrix} 1 & 0 \\ 0 & e^{\frac{2\pi i}{k}} \end{bmatrix}$$

# Implementation and comparison with iFFT



# Estimation of the QFT running time

- Add up the delays on the longest path of the compiled circuit
- Single qubit gate: 10 ns, two qubit gate: 100 ns

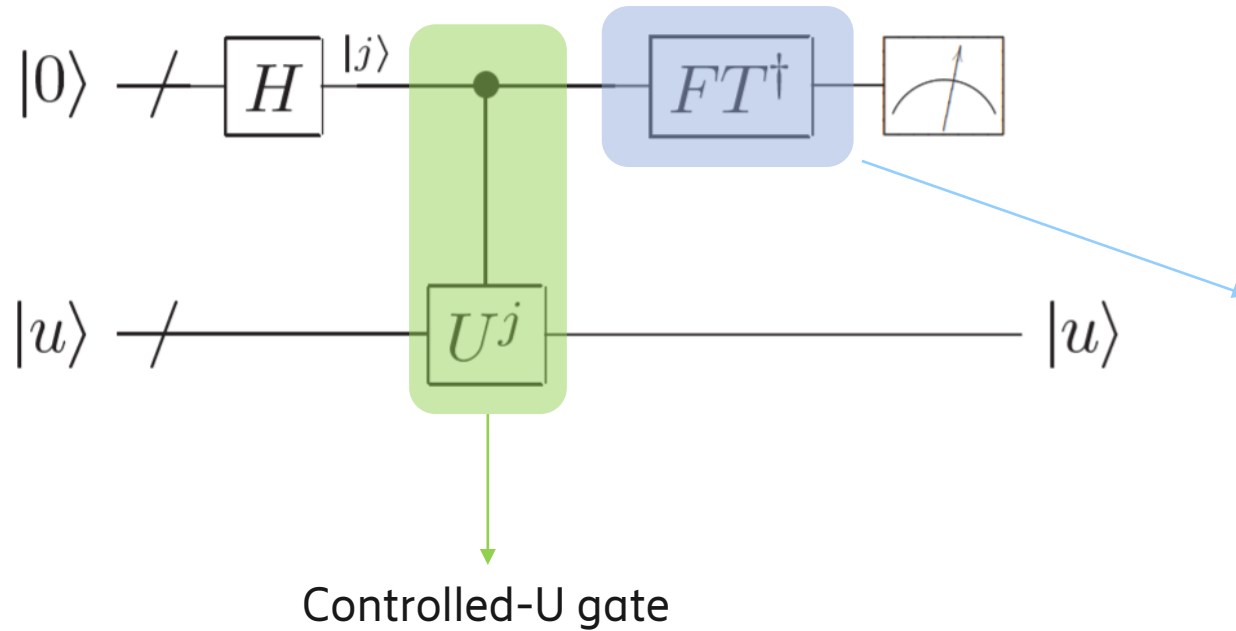


- e.g. For 3-qubit circuit, the estimated running time is:
  - 6 single qubit gates + 1 two qubits gate (SWAP gate) = 160ns
- Computational cost for  $n$  qubits (or for  $2^n$  bits): QFT  $\Theta(n^2)$ , FFT  $\Theta(n \cdot 2^n)$

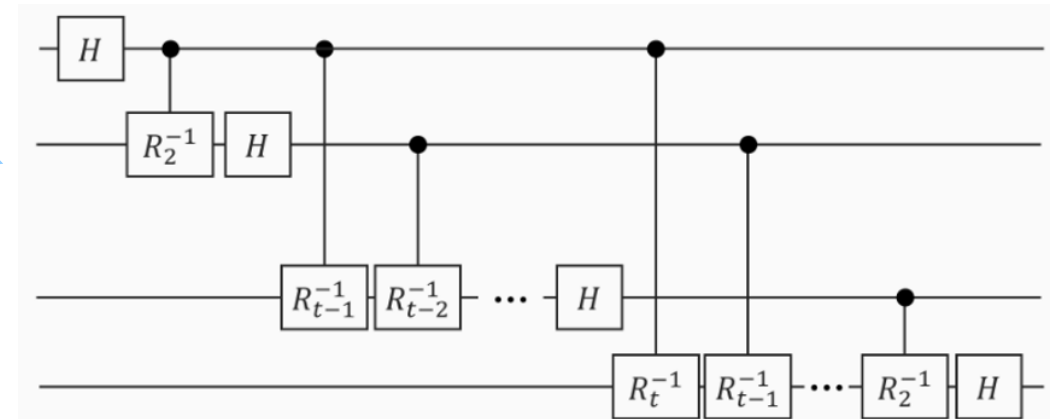


# Application: Quantum Phase Estimation

Quantum phase estimation: to calculate the eigenvalue of  $U$



Inverse quantum Fourier transform:

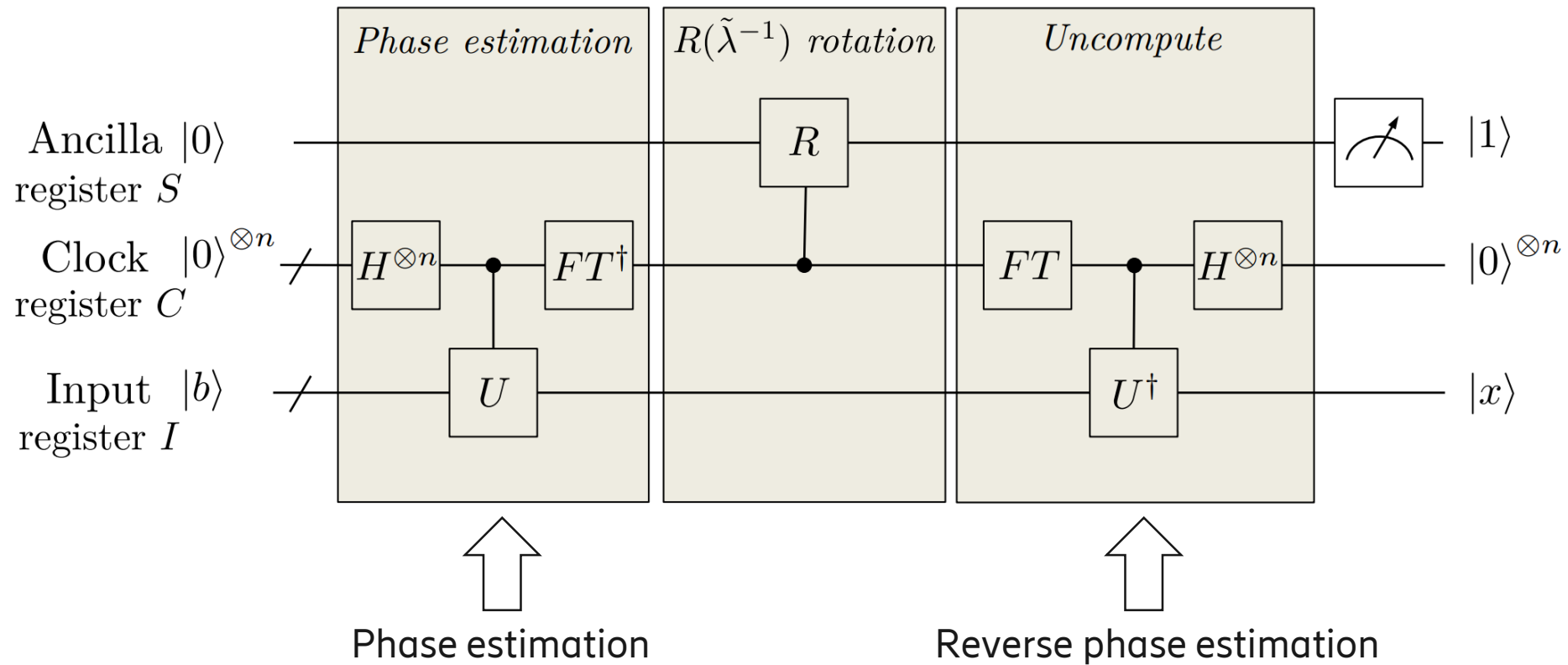


$$|j\rangle \rightarrow \frac{1}{2^{\frac{n}{2}}} \sum_{k=0}^{2^n-1} e^{-2\pi i j k / 2^n} |k\rangle$$

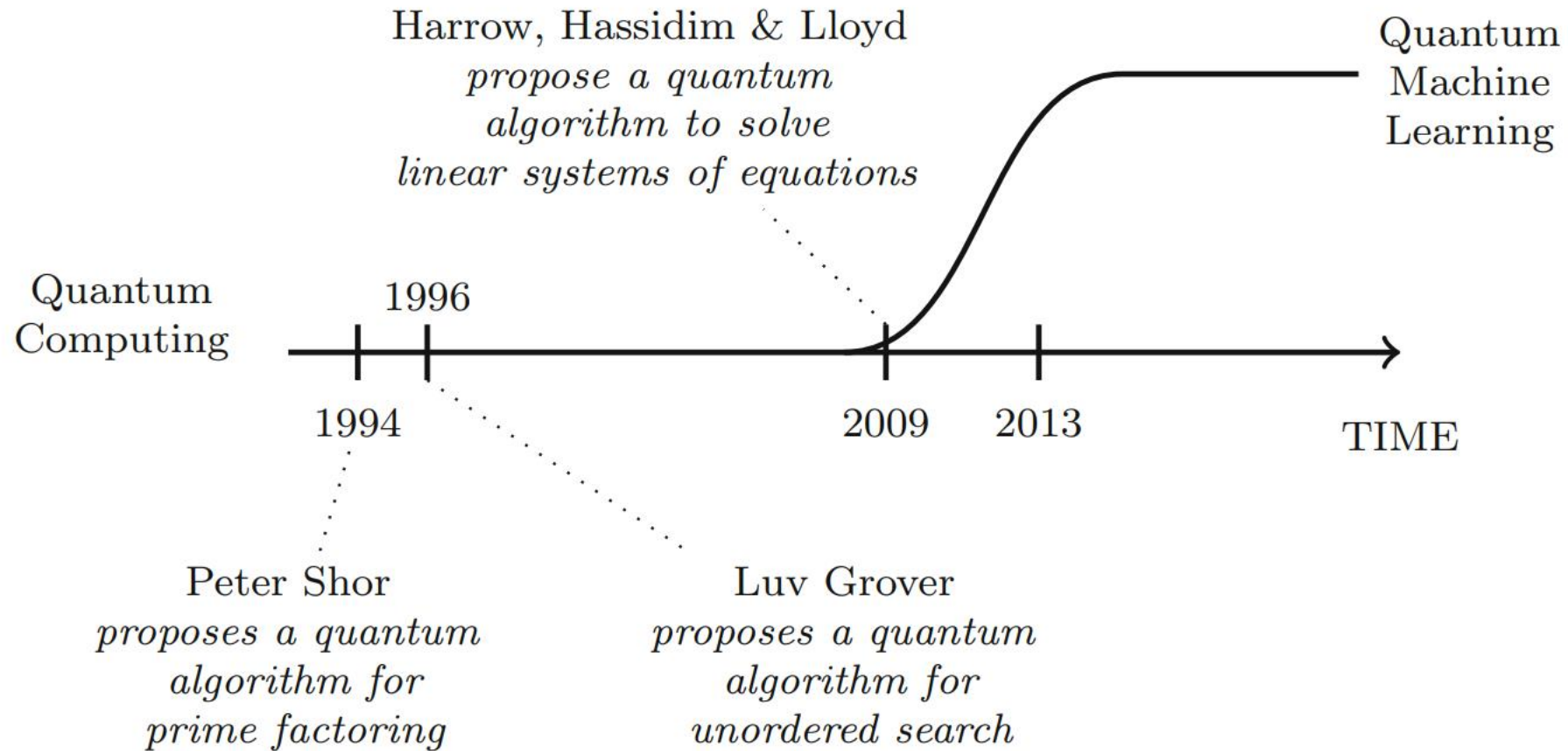
$$R_k^{-1} \equiv \begin{bmatrix} 1 & 0 \\ 0 & e^{-\frac{2\pi i}{k}} \end{bmatrix}$$

# Application: HHL Algorithm- Quantum algorithm for linear systems of equations

Solve  $\vec{A}x = \vec{b}$  using a quantum computer



# Application: HHL Algorithm- Milestone for QML



Usage: quantum machine learning algorithms including bayesian inference, qSVM, qPCA...

# Materials

- Books:

- Quantum Computation and Quantum Information, 2002

- Quantum Machine Learning: What Quantum Computing Means to Data Mining, 2014

- Supervised Learning with Quantum Computers, 2018

- Online courses:

- edX & University of Toronto: Quantum Machine Learning

- edX & TU Delft: Quantum Cryptography