26/06/2023

School of informatics of
Barcelona - UPC

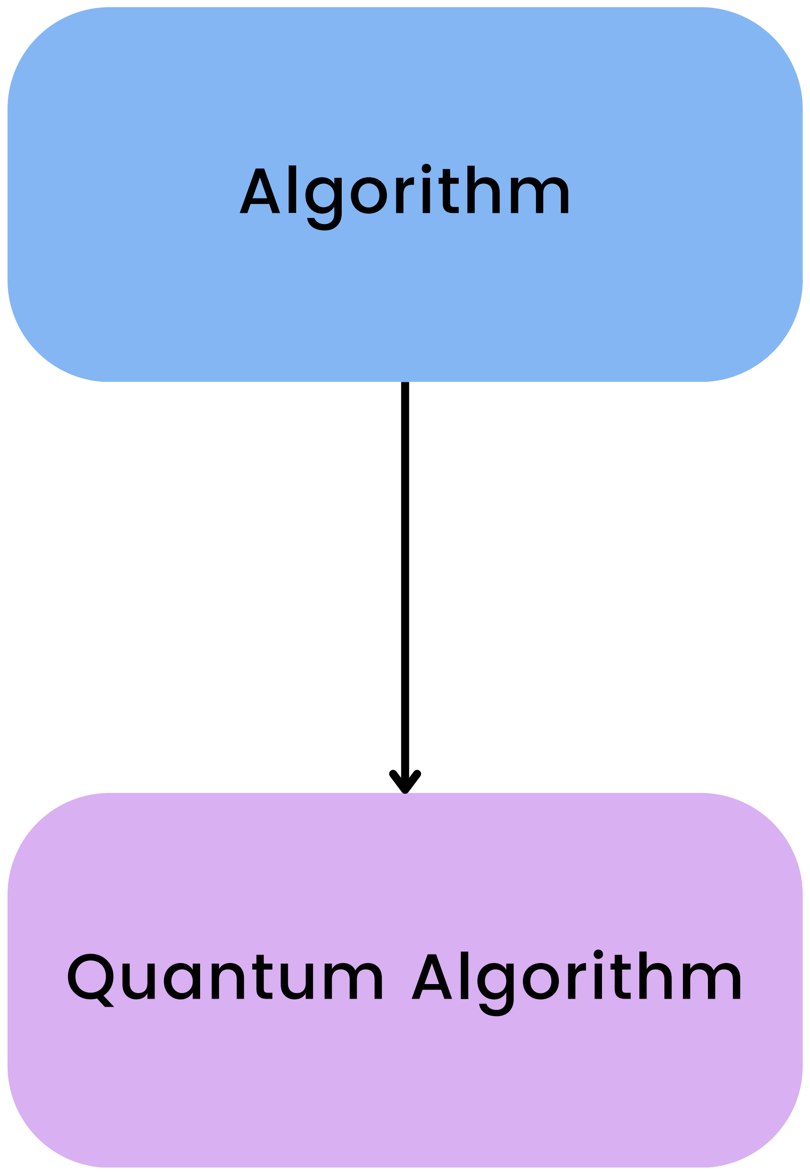
SUPERVISORS:
JORDI DELGADO PIN (UPC)
ALBA CERVERA LIERTA (BSC-CNS)

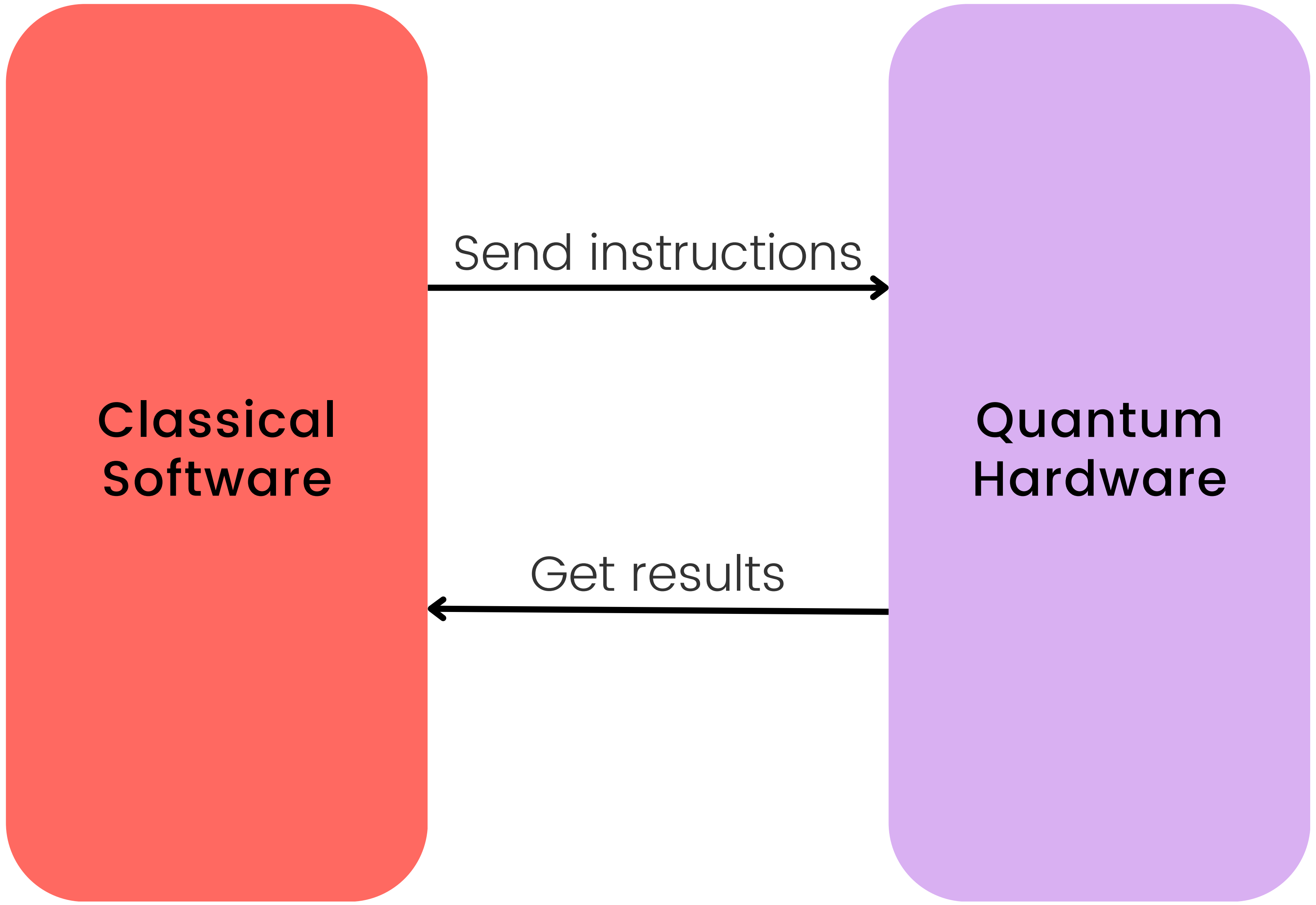
ALEXANDRE ALEMANY
ORFILA

Harrow-Hassidim-Lloyd algorithm:
Study and implementation using Qibo

FINAL PROJECT DEFENCE

I Motivation





II Introduction

Linear Solver Problem (LSP)

$$\begin{cases} 3x + 5y + z = 0 \\ 7x - 2y + 4z = 0 \\ -6x + 3y + 2z = 0 \end{cases}$$

Fig 1: system of linear equations

$$\begin{bmatrix} 3 & 5 & 1 \\ 7 & -2 & 4 \\ -6 & 3 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Fig 2: System represented using matrices

II Introduction

The HHL Algorthm and the Quantum LSP (QLSP)

Quantum algorithm for linear systems of equations

Aram W. Harrow,[1] Avinatan Hassidim,[2] and Seth Lloyd[3]

[1] *Department of Mathematics, University of Bristol, Bristol, BS8 1TW, U.K.*
[2] *MIT - Research Laboratory for Electronics, Cambridge, MA 02139, USA*
[3] *MIT - Research Laboratory for Electronics and Department of Mechanical Engineering, Cambridge, MA 02139, USA*

Solving linear systems of equations is a common problem that arises both on its own and as a subroutine in more complex problems: given a matrix A and a vector $\vec{b}$, find a vector $\vec{x}$ such that $A\vec{x} = \vec{b}$. We consider the case where one doesn't need to know the solution $\vec{x}$ itself, but rather an approximation of the expectation value of some operator associated with $\vec{x}$, e.g., $\vec{x}^\dagger M \vec{x}$ for some matrix M. In this case, when A is sparse, $N \times N$ and has condition number κ, classical algorithms can find $\vec{x}$ and estimate $\vec{x}^\dagger M \vec{x}$ in $\tilde{O}(N\sqrt{\kappa})$ time. Here, we exhibit a quantum algorithm for this task that runs in $\mathrm{poly}(\log N, \kappa)$ time, an exponential improvement over the best classical algorithm.

Fig 3: Original paper of the HHL algorithm [1]

III Project Tasks Overview

- Understanding of quantum circuits and quantum computing notation as well as the basic mathematical procedures used in both.
- Review and understanding of different papers (original papers and enhancement proposals) about the quantum algorithms explained in this project.
- Review of the mathematical procedures required to justify the correctness.
- Study of time complexity of the quantum algorithms.
- The general implementation for the Quantum Fourier Transform and the Quantum Phase Estimation using Qibo.
- The implementation of an example of the HHL circuit using Qibo explaining it step by step.
- The study of the results of the circuits using the simulator backend that provides Qibo.

IV Quantum Fourier Transform

Fourier Transform and Discrete Fourier Transform

$$F(k) = \int_{-\infty}^{\infty} f(x) e^{-i2\pi kx} dx$$

Fig 4: Fourier transform formula

$$X_k = \sum_{n=0}^{N-1} x_n e^{-i2\pi kn/N}$$

Fig 5: Discrete Fourier transform formula

IV Quantum Fourier Transform

From DFT to QFT

$$F_{jk} = \frac{1}{\sqrt{N}} e^{-i2\pi kj/N}, \quad j, k = 0, 1, \ldots, N-1.$$

Fig 6: Definition of the matrix

$$F = \frac{1}{\sqrt{N}} \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega & \omega^2 & \cdots & \omega^{N-1} \\ 1 & \omega^2 & \omega^4 & \cdots & \omega^{2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega^{N-1} & \omega^{2(N-1)} & \cdots & \omega^{(N-1)(N-1)} \end{pmatrix}$$

Fig 7: The matrix operator $\omega = e^{-i2\pi/N}$

IV Quantum Fourier Transform

From DFT to QFT

$$|k\rangle \mapsto F_N|k\rangle = \frac{1}{\sqrt{N}}\sum_{j=0}^{N-1}\omega_N^{jk}|j\rangle, \quad j,k = 0,1,\dots,N-1$$

Fig 8: Formula of the Quantum Fourier Transform

$$\mathbf{y} = F\mathbf{x} = \frac{1}{\sqrt{N}}\begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega & \omega^2 & \cdots & \omega^{N-1} \\ 1 & \omega^2 & \omega^4 & \cdots & \omega^{2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega^{N-1} & \omega^{2(N-1)} & \cdots & \omega^{(N-1)(N-1)} \end{pmatrix}\begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_{N-1} \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \\ y_{N-1} \end{pmatrix}$$

$$\omega = e^{-i2\pi/N}$$

Fig 7: Transforming with the matrix operator

IV Quantum Fourier Transform
The quantum circuit

$$F_N|k\rangle = \bigotimes_{l=1}^{n} \frac{1}{\sqrt{2}}(|0\rangle + e^{i2\pi k/2^l}|1\rangle)$$

Fig 10: QFT formula using tensor products

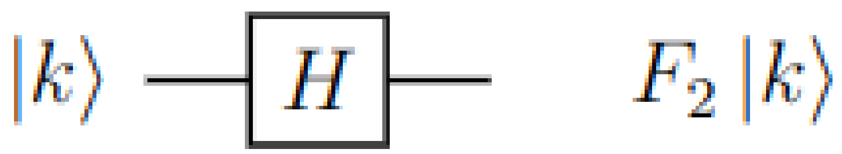


Fig 11: Quantum circuit for the QFT in one qubit

- for $N = 2$:

$$F_2|k_1\rangle = \bigotimes_{l=1}^{1} \frac{1}{\sqrt{2}}(|0\rangle + e^{i2\pi k/2^l}|1\rangle) = \frac{1}{\sqrt{2}}(|0\rangle + e^{i\pi k}|1\rangle) = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

Fig 12: Example for one qubit

IV Quantum Fourier Transform

The quantum circuit

$$F_N|k\rangle = \bigotimes_{l=1}^{n} \frac{1}{\sqrt{2}}(|0\rangle + e^{i2\pi k/2^l}|1\rangle)$$

Fig 10: QFT formula using tensor products

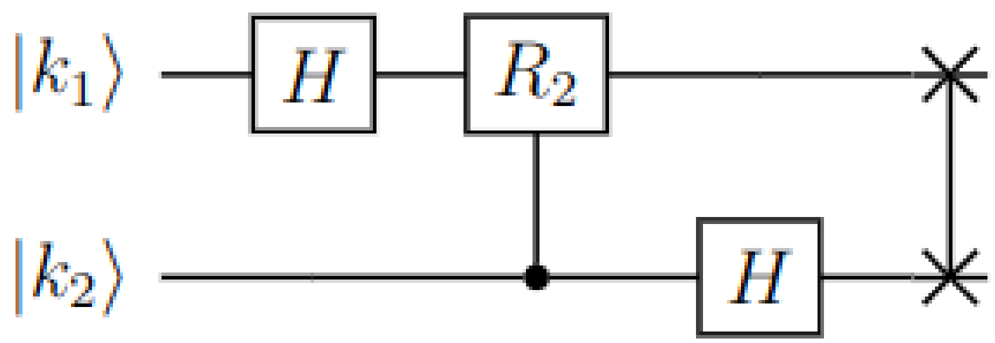


Fig 13: Quantum circuit for the QFT in two qubits

- for $N = 4$

$$F_4|k_1k_2\rangle = \bigotimes_{l=1}^{2} \frac{1}{\sqrt{2}}(|0\rangle + e^{i2\pi k/2^l}|1\rangle) = \frac{1}{\sqrt{2}}(|0\rangle + e^{i2\pi 0.k_2}|1\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle + e^{i2\pi 0.k_1k_2}|1\rangle)$$

$$= \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle + e^{i2\pi 0.k_1k_2}|1\rangle) \tag{14}$$

Fig 14: Example for two qubits

IV Quantum Fourier Transform

Time complexity

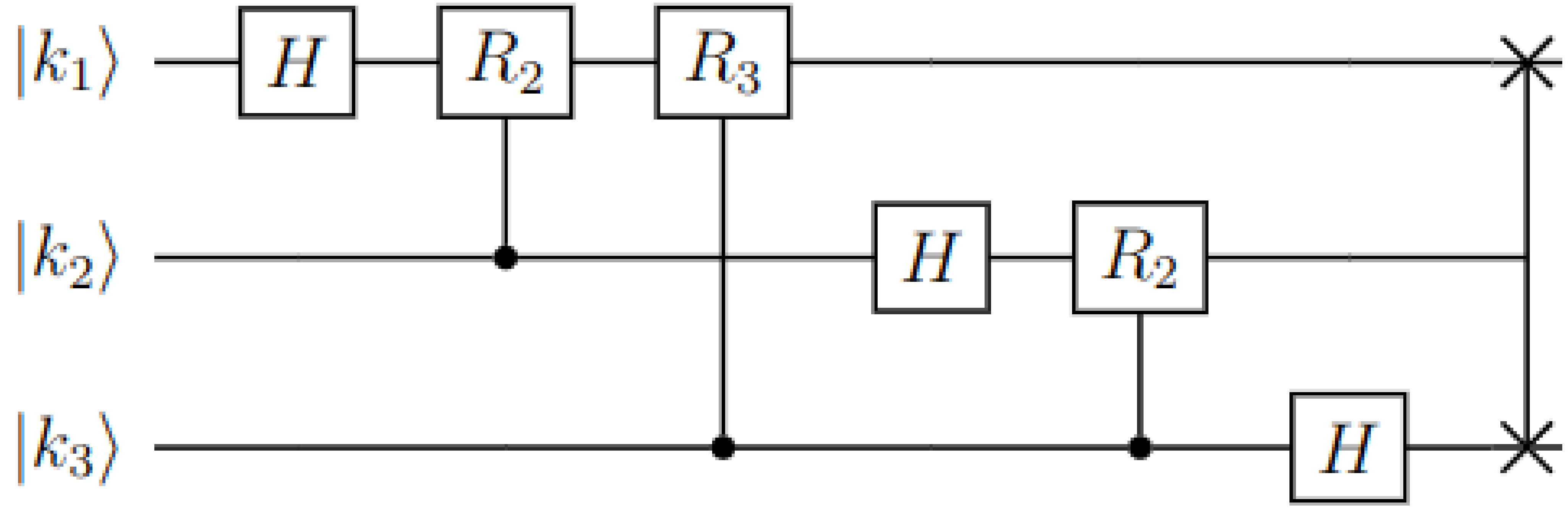


Fig 15: Quantum circuit for the QFT in three qubits

time cost of the QFT: $O(n^2)$

V Quantum Phase Estimation

Problem Statement

Let U be a unitary operator with eigenvalue $e^{2\pi i\phi}$. We want to find the phase of U.

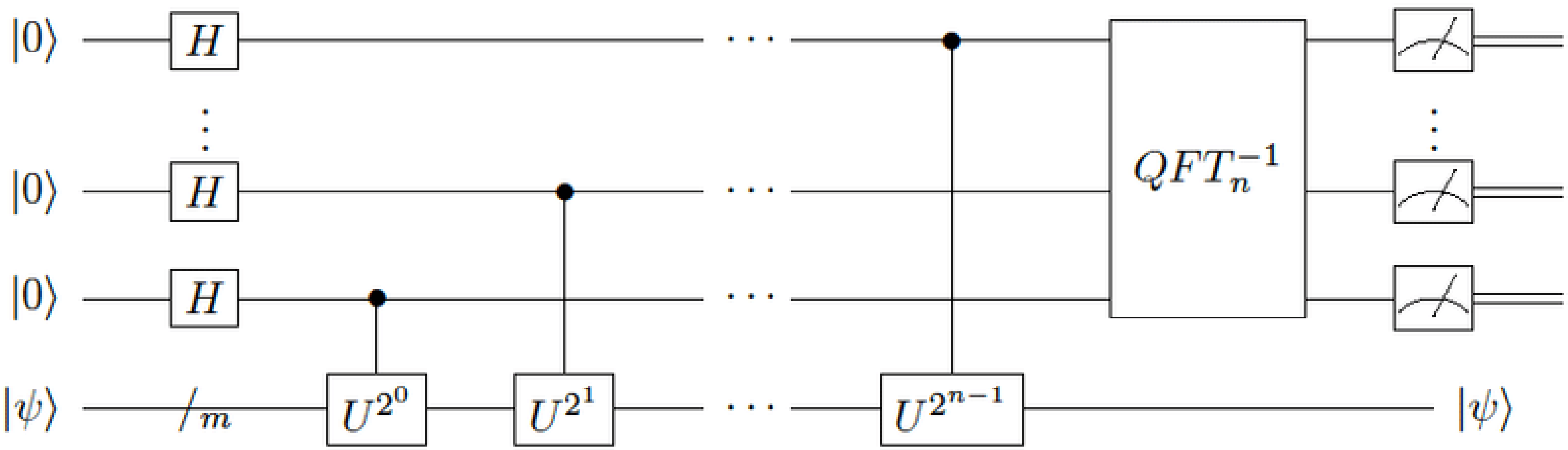


Fig 16: Quantum Phase Estimation Circuit[2]

V Quantum Phase Estimation

Problem Statement

Let U be a unitary operator with eigenvalue $e^{2\pi i\phi}$. We want to find the phase of U.

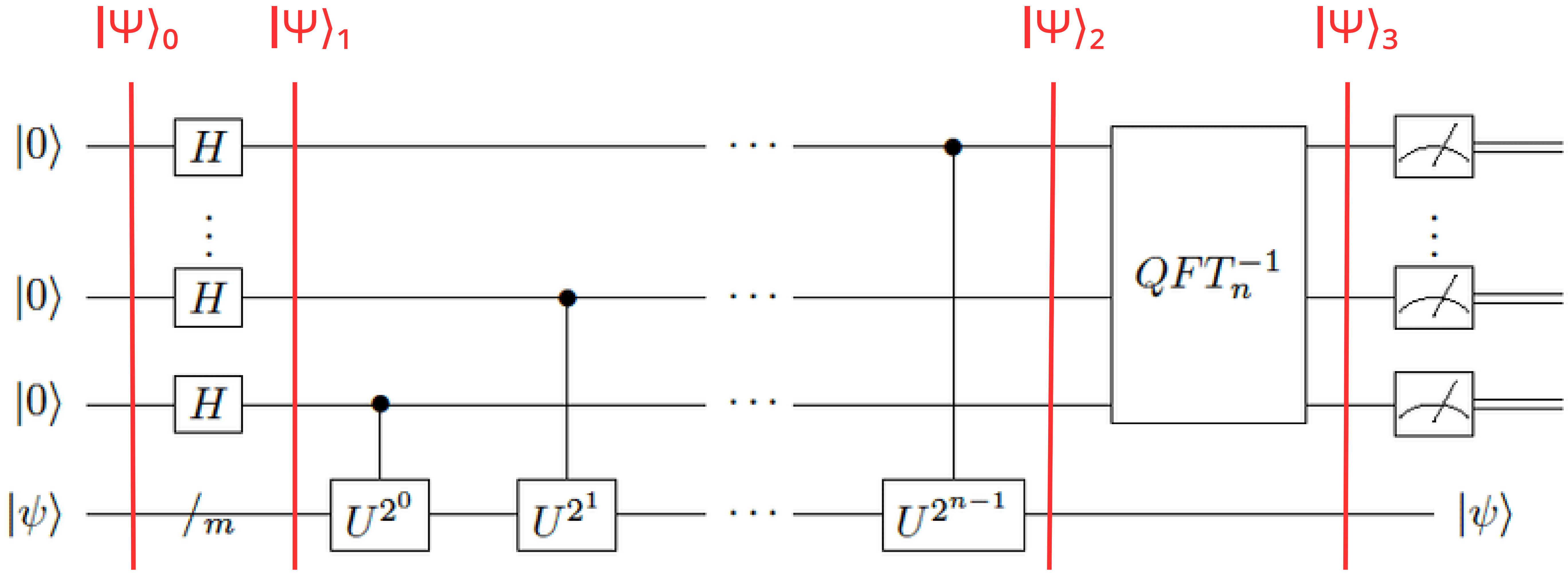


Fig 16: Quantum Phase Estimation circuit[2]

V Quantum Phase Estimation

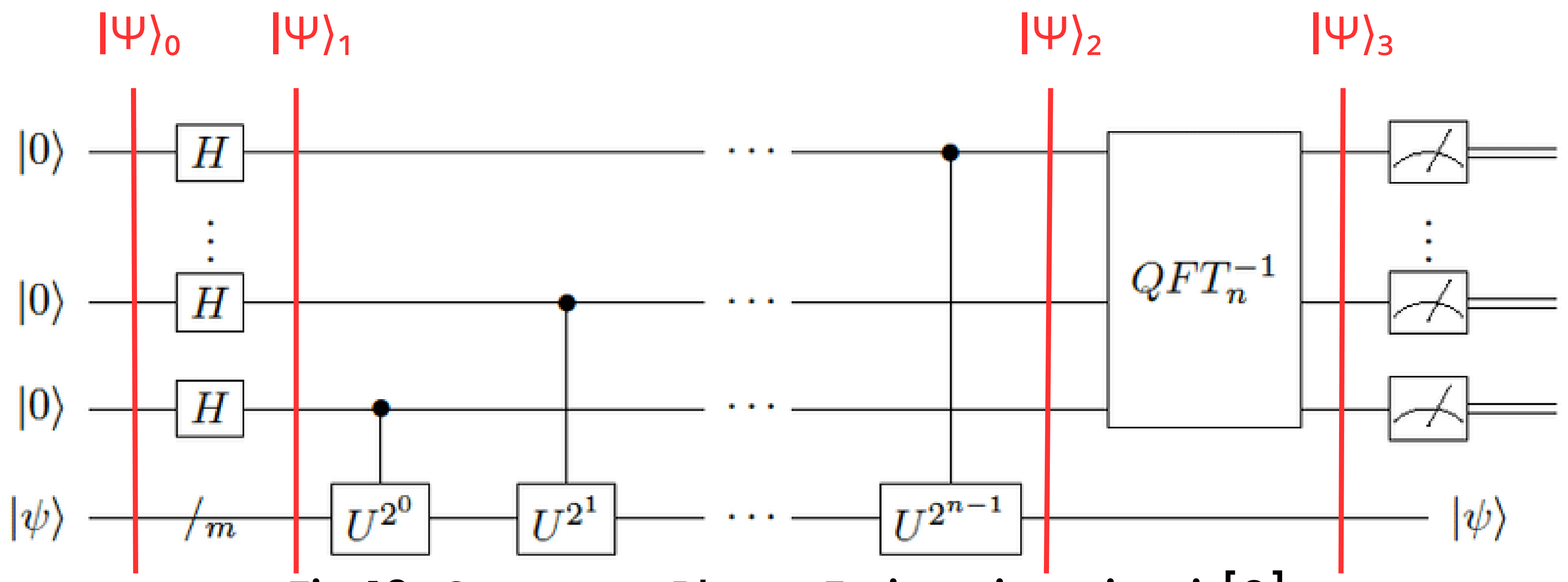


Fig 16: Quantum Phase Estimation circuit[2]

$$|\psi_0\rangle = |0\rangle^{\otimes m} |\psi\rangle \longrightarrow |\psi_1\rangle = H^{\otimes m} |0\rangle^{\otimes m} |\psi\rangle = \frac{1}{2^{\frac{m}{2}}}(|0\rangle + |1\rangle)^{\otimes m} |\psi\rangle$$

Fig 17: The Hadamard gates are applied to the estimation register

V Quantum Phase Estimation

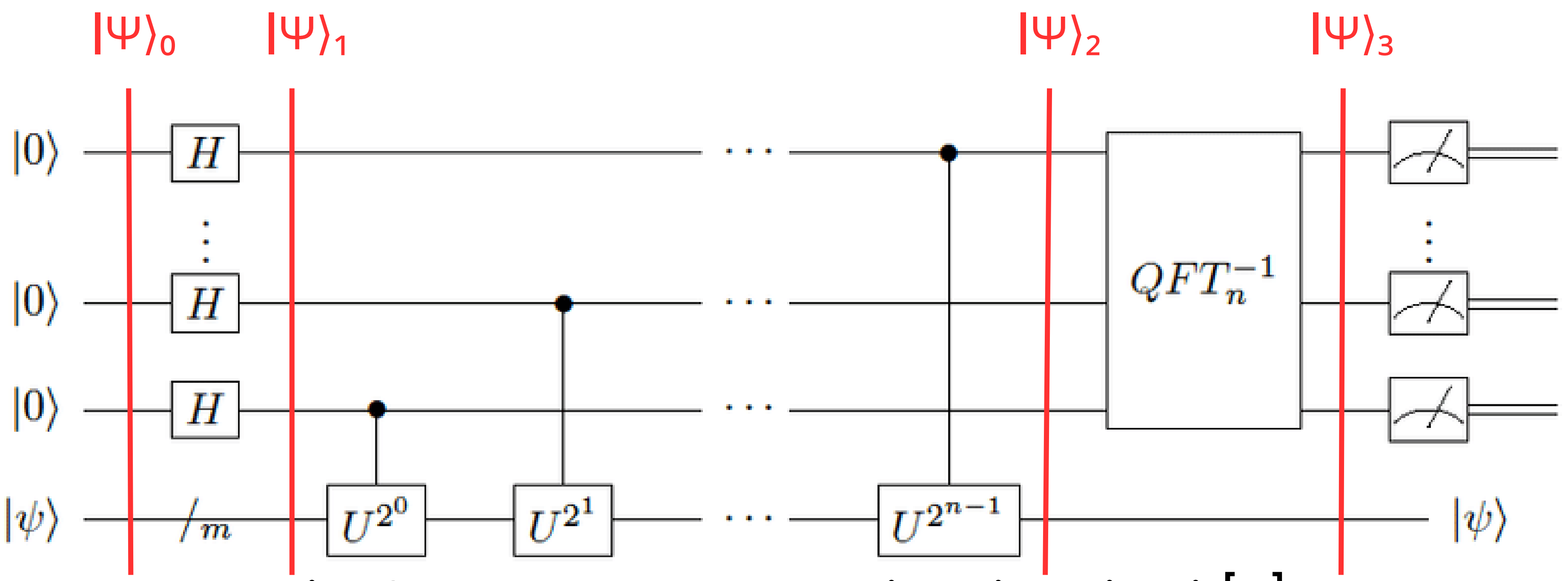


Fig 16: Quantum Phase Estimation circuit[2]

$$U^{2^j}\left|\psi\right\rangle = U^{2^j-1}U\left|\psi\right\rangle = U^{2^j-1}e^{2\pi i\phi}\left|\psi\right\rangle = e^{2\pi i 2^j\phi}\left|\psi\right\rangle$$

$$|\psi_2\rangle = \frac{1}{2^{\frac{m}{2}}}\left(|0\rangle + e^{2\pi i\phi 2^{m-1}}|1\rangle\right)\otimes\cdots\otimes\left(|0\rangle + e^{2\pi i\phi 2^{0}}|1\rangle\right)\otimes|\psi\rangle \ = \frac{1}{2^{\frac{m}{2}}}\sum_{k=0}^{2^m-1} e^{2\pi i\phi k}|k\rangle\otimes|\psi\rangle$$

Fig 18: The U-controlled gates are applied

V Quantum Phase Estimation

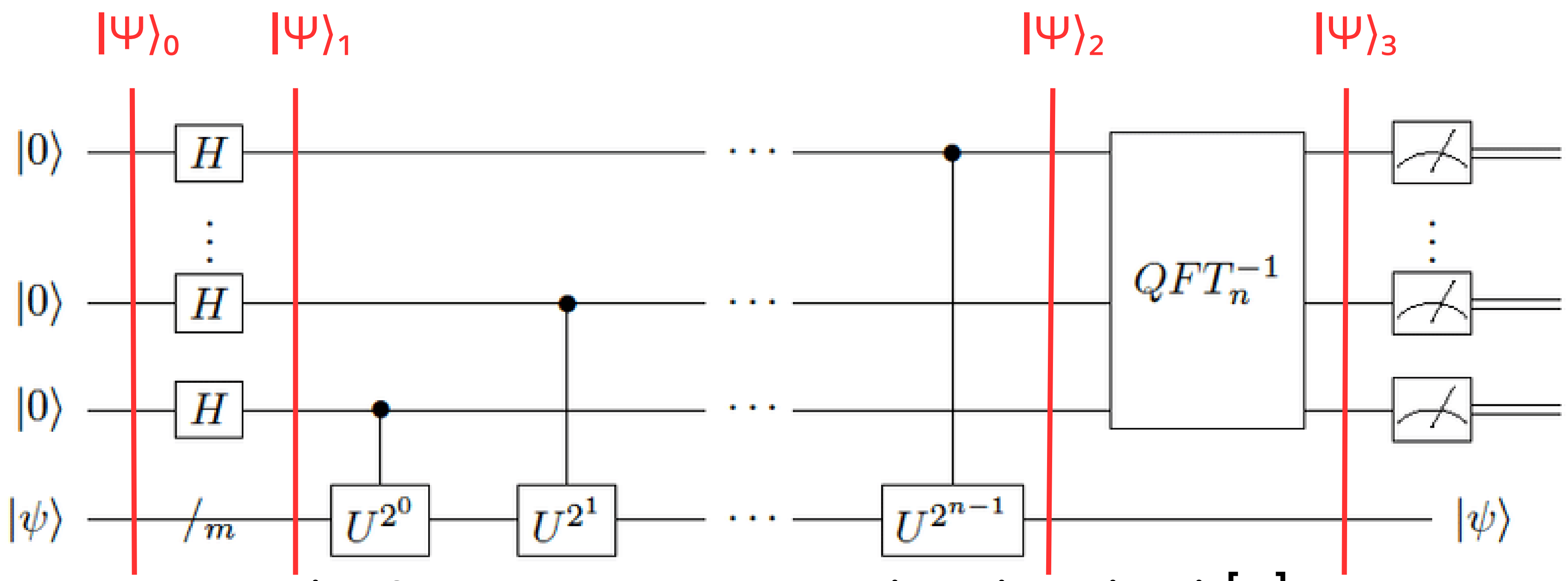


Fig 16: Quantum Phase Estimation circuit[2]

$$\frac{1}{2^{\frac{m}{2}}} \sum_{k=0}^{2^m-1} e^{2\pi i \phi k} |k\rangle \otimes |\psi\rangle \longrightarrow |\psi_3\rangle = |2^m \phi\rangle \otimes |\psi\rangle$$

Fig 19: Applying the IQFT

V Quantum Phase Estimation

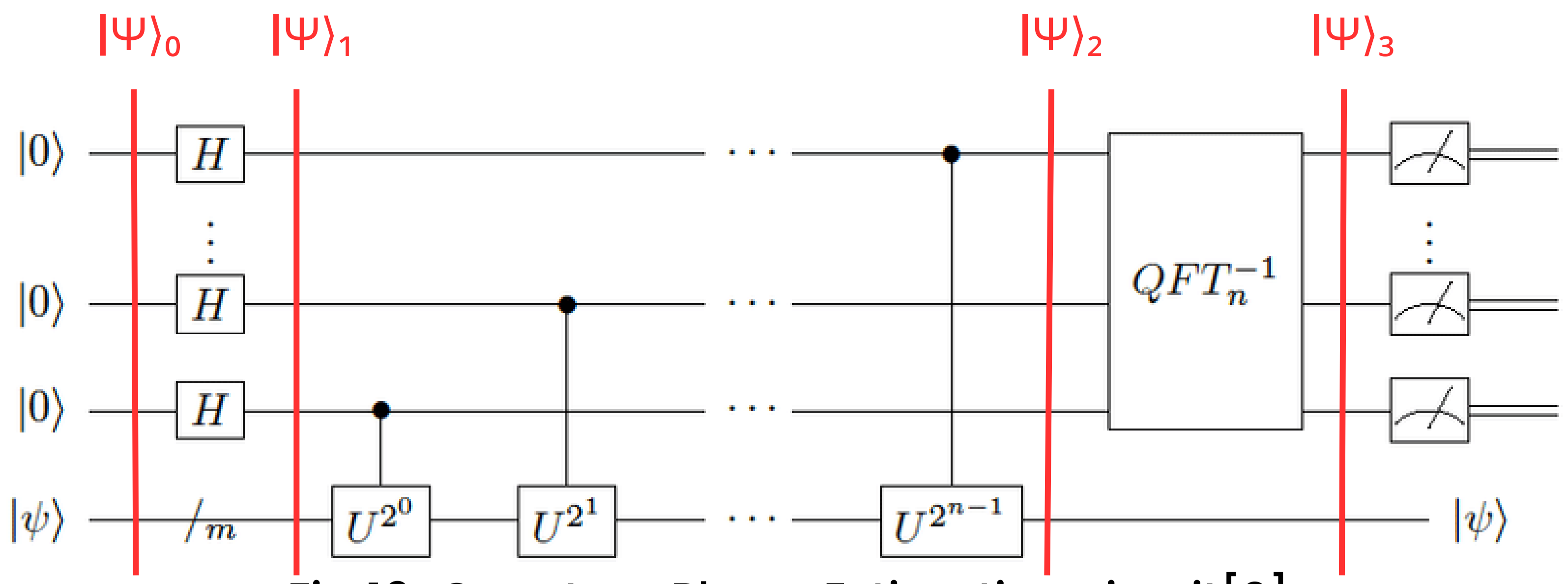


Fig 16: Quantum Phase Estimation circuit[2]

Probability of measuring ϕ	
if ϕ is a fraction of a power of 2: 100%	if ϕ is not a fraction of a power of 2: $4/\pi^2$

Fig 20: Probabilities of measuring ϕ

V Quantum Phase Estimation

Time complexity

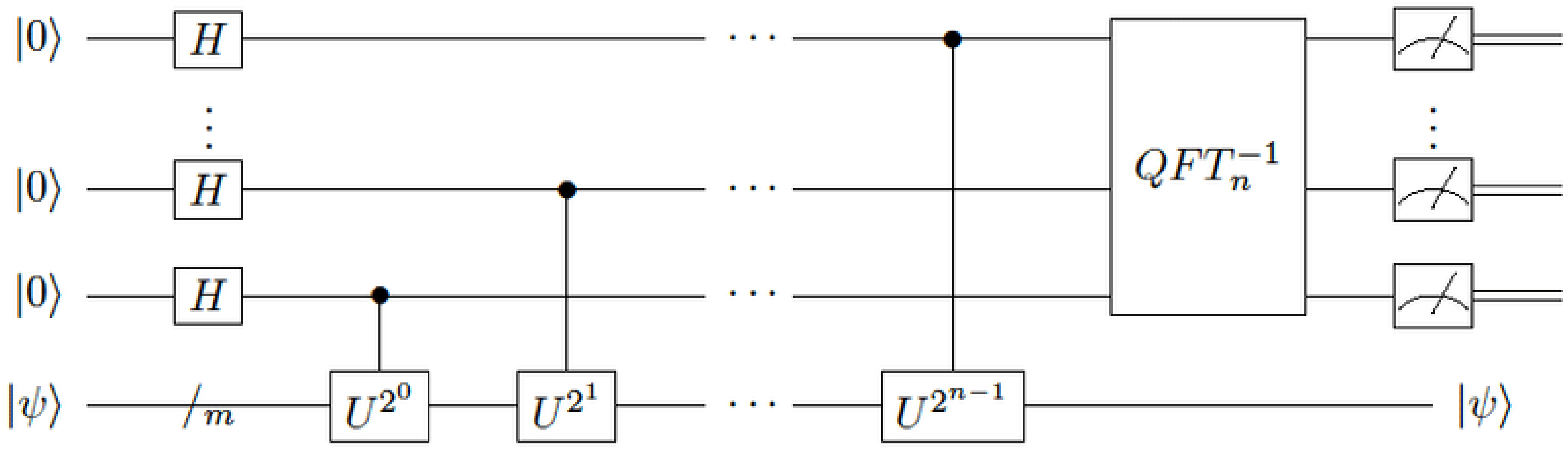


Fig 16: Quantum Phase Estimation circuit[2]

The cost of the QPE is $O(n^2)$

Accepting some error is $O(n + \log(1/e)) = O(n)$

VI Harrow-Hassidim-Lloyd algorithm:
The Quantum Linear solver problem

LSP	QLSP
$Ax = b$	$A\|x\rangle = \|b\rangle$
The vector x is found	A state similar to $\|x\rangle$ is found

Fig 21: Differences between QLSP and LSP

$$A = \begin{pmatrix} 1 & -\frac{1}{3} \\ -\frac{1}{3} & 1 \end{pmatrix} \qquad \vec{b} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \qquad \vec{x} = \begin{pmatrix} \frac{3}{8} \\ \frac{9}{8} \end{pmatrix}$$

Fig 22: Example of LSP

VI Harrow-Hassidim-Lloyd algorithm:
Assumptions and set up of the algorithm

Assumptions:

- $|b\rangle$ is loaded efficiently
- A is hermitian and invertible

Set up:

- $|b\rangle = \sum_i b_i |u_i\rangle$
- $A = \sum_i \lambda_i |u_i\rangle \langle u_i|$
- $A^{-1} = \sum_i \frac{1}{\lambda_i} |u_i\rangle \langle u_i|$
- $|x\rangle = \sum_i \frac{b_i}{\lambda_i} |u_i\rangle$

VI Harrow-Hassidim-Lloyd algorithm:
The algorithm

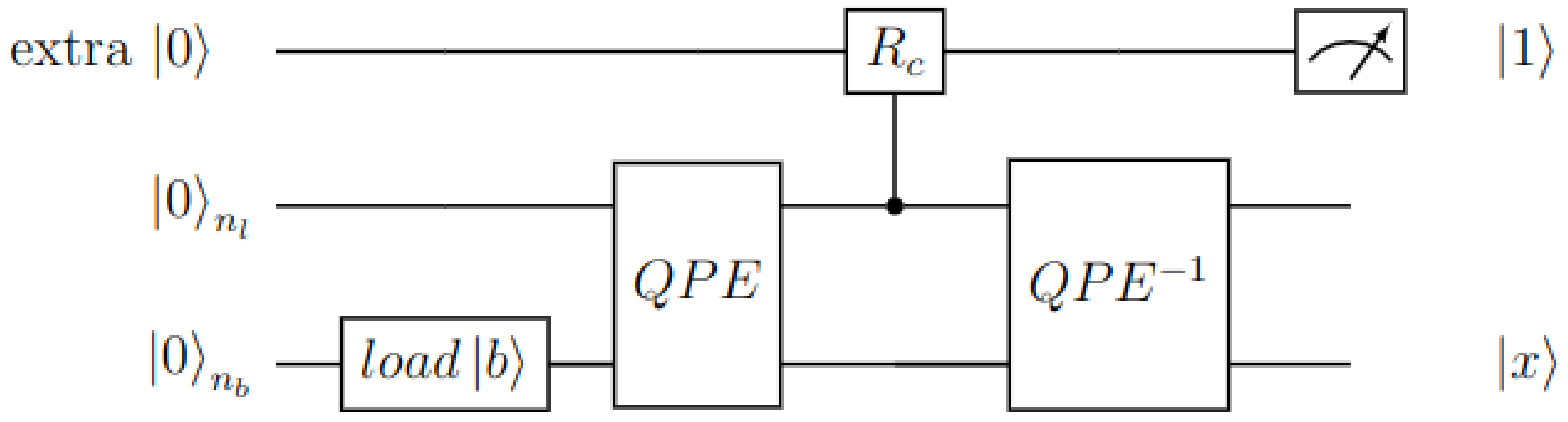


Fig 23: Circuit of the HHL algorithm

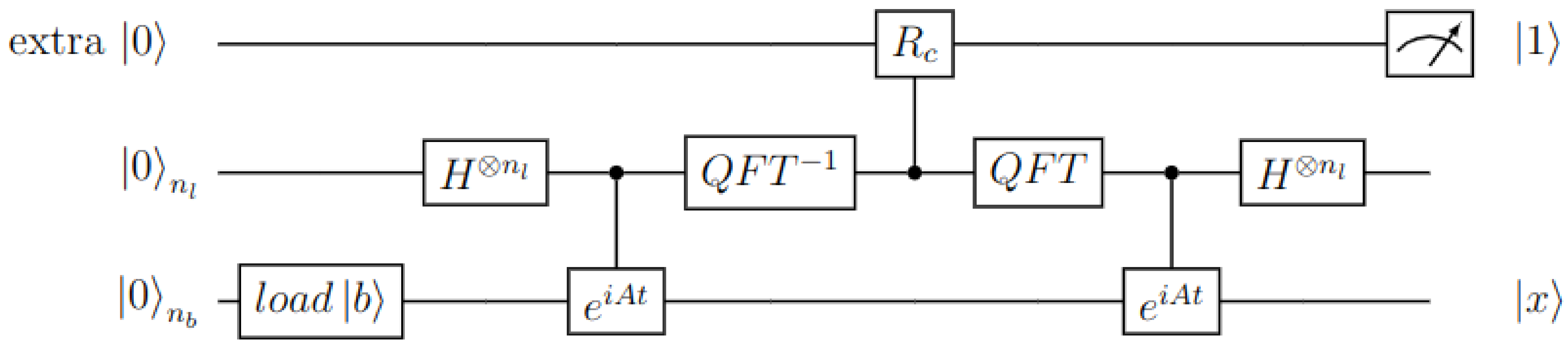


Fig 24: Circuit of the HHL algorithm detailing the QPE

VI Harrow-Hassidim-Lloyd algorithm:

The algorithm

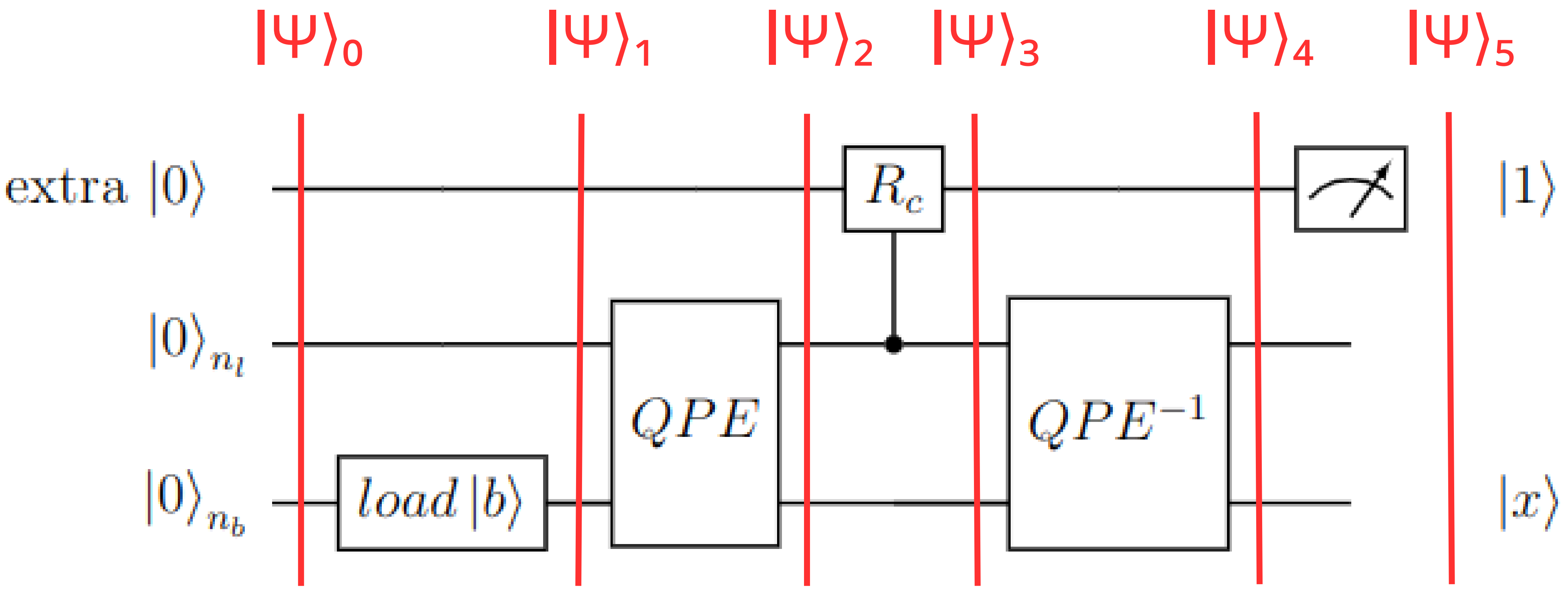


Fig 23: Circuit of the HHL algorithm

VI Harrow-Hassidim-Lloyd algorithm:
The algorithm

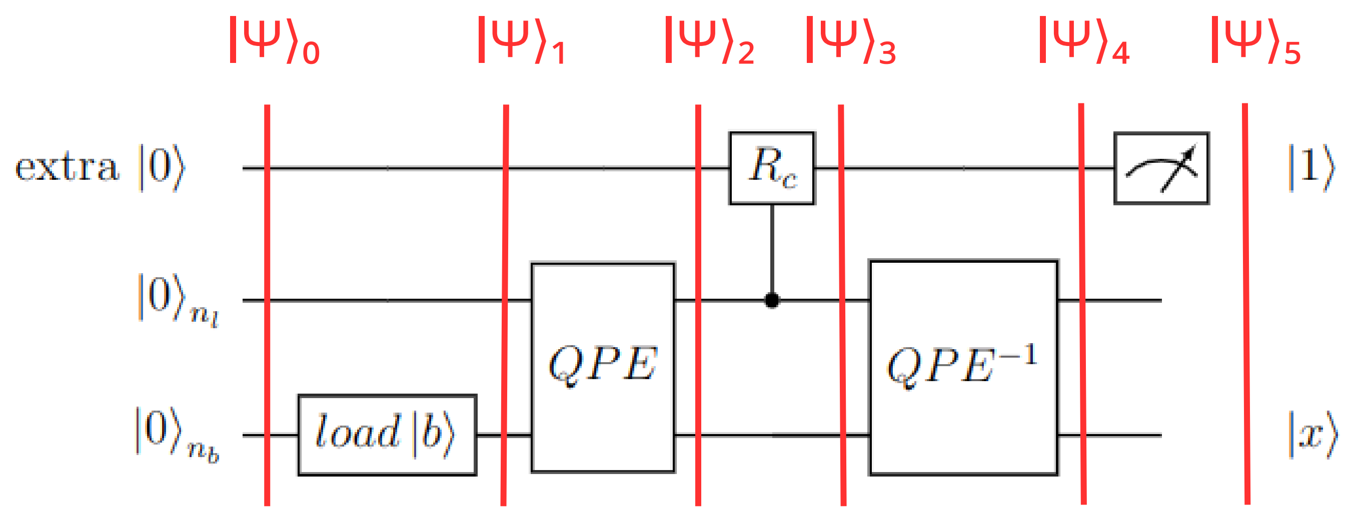


Fig 23: Circuit of the HHL algorithm

$$|\psi_0\rangle = |0\rangle_{n_b} |0\rangle_{n_l} |0\rangle \longmapsto |\psi_1\rangle = |b\rangle_{n_b} |0\rangle_{n_l} |0\rangle = \sum_i b_i |u_i\rangle_{n_b} |0\rangle_{n_l} |0\rangle$$

Fig 25: |b⟩ is loaded to the register

VI Harrow-Hassidim-Lloyd algorithm:
The algorithm

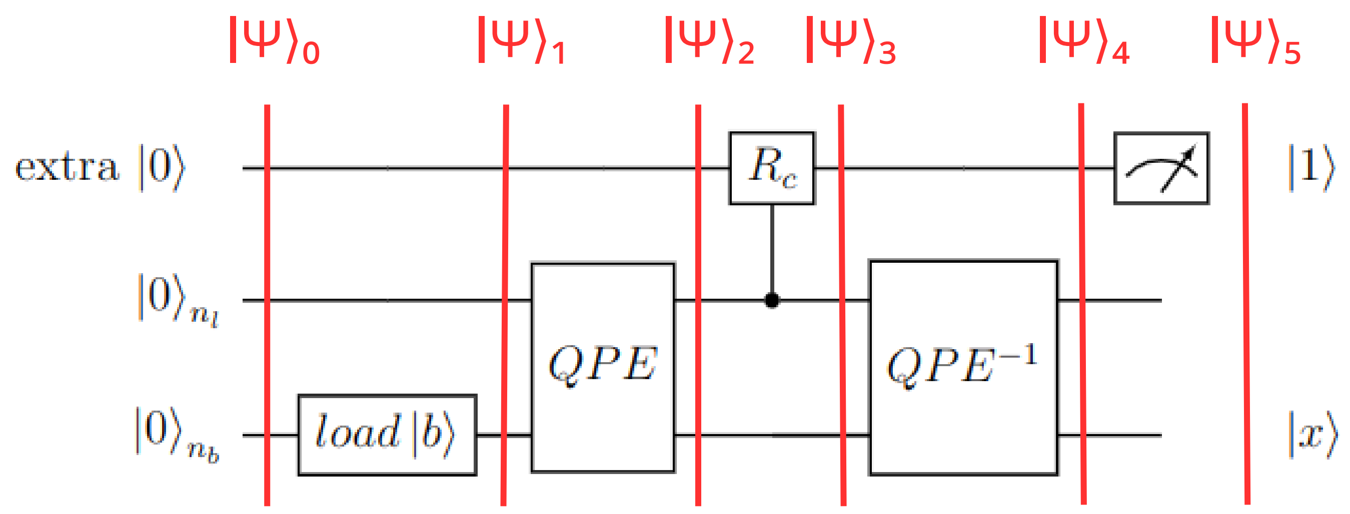


Fig 23: Circuit of the HHL algorithm

$$|\psi_1\rangle \longmapsto |\psi_2\rangle = \sum_i b_i \, |u_i\rangle_{n_b} \, |\lambda_i\rangle_{n_l} \, |0\rangle$$

Fig 26: QPE is applied to get the eigenvalues of A

VI Harrow-Hassidim-Lloyd algorithm:
The algorithm

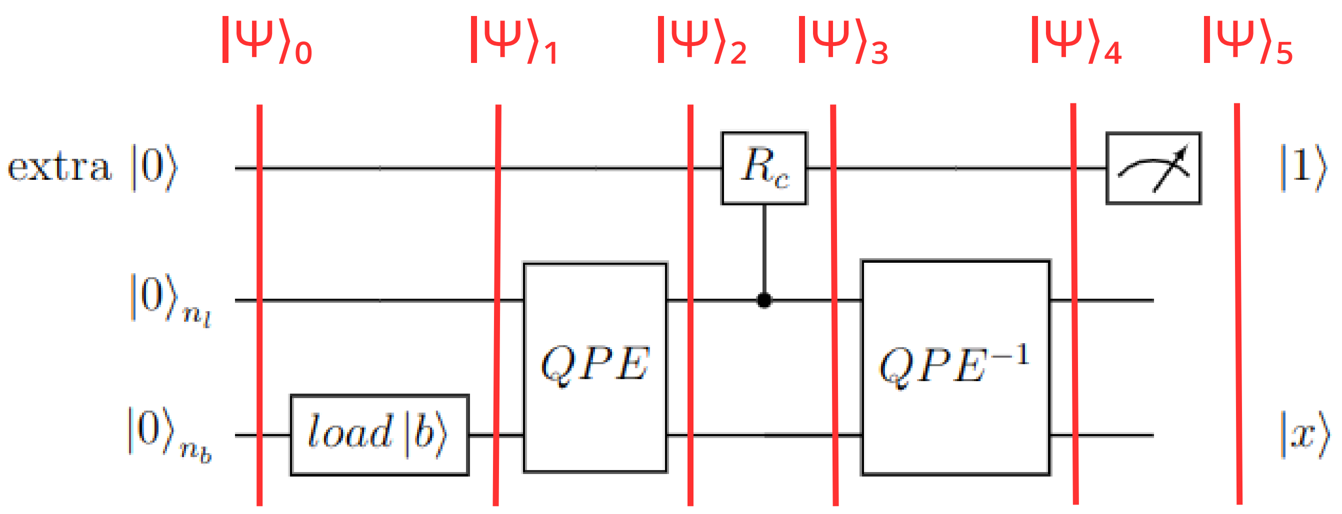


Fig 23: Circuit of the HHL algorithm

$$|\psi_2\rangle \longmapsto |\psi_3\rangle = \sum_i b_i \, |u_i\rangle_{n_b} \, |\lambda_i\rangle_{n_l} \left(\sqrt{1 - \frac{C^2}{\lambda_i^2}} \, |0\rangle + \frac{C}{\lambda_i} \, |1\rangle \right)$$

Fig 27: The conditional rotation is applied

VI Harrow-Hassidim-Lloyd algorithm:
The algorithm

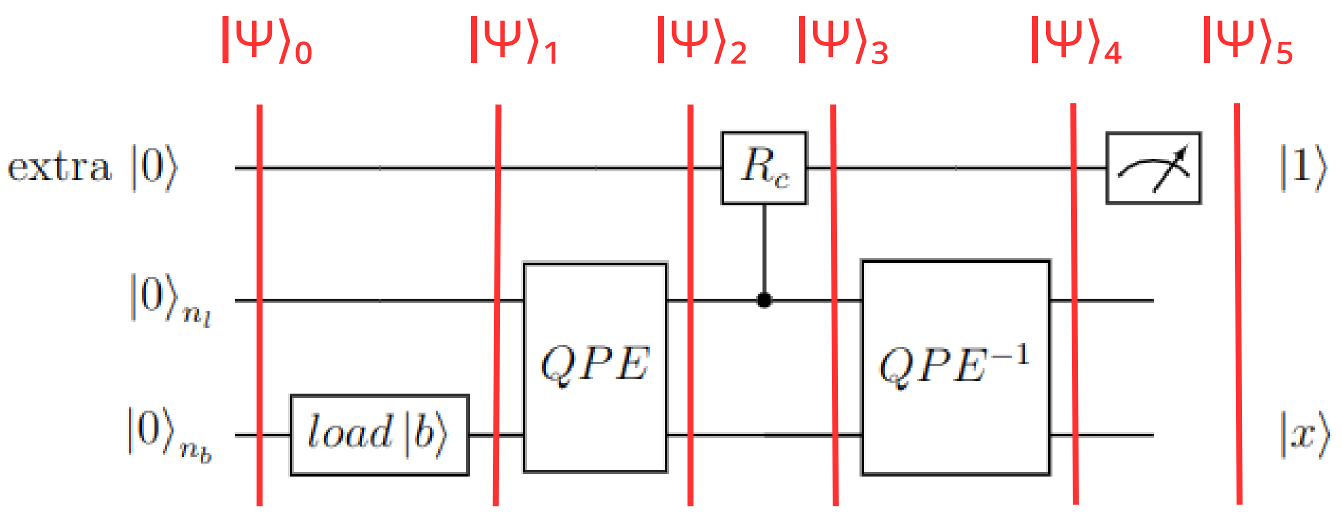


Fig 23: Circuit of the HHL algorithm

$$|\psi_3\rangle \longmapsto |\psi_4\rangle = \sum_i b_i \, |u_i\rangle_{n_b} |0\rangle_{n_l} \left(\sqrt{1 - \frac{C^2}{\lambda_i^2}} \, |0\rangle + \frac{C}{\lambda_i} |1\rangle \right)$$

Fig 28: IQPE is applied

VI Harrow-Hassidim-Lloyd algorithm:
The algorithm

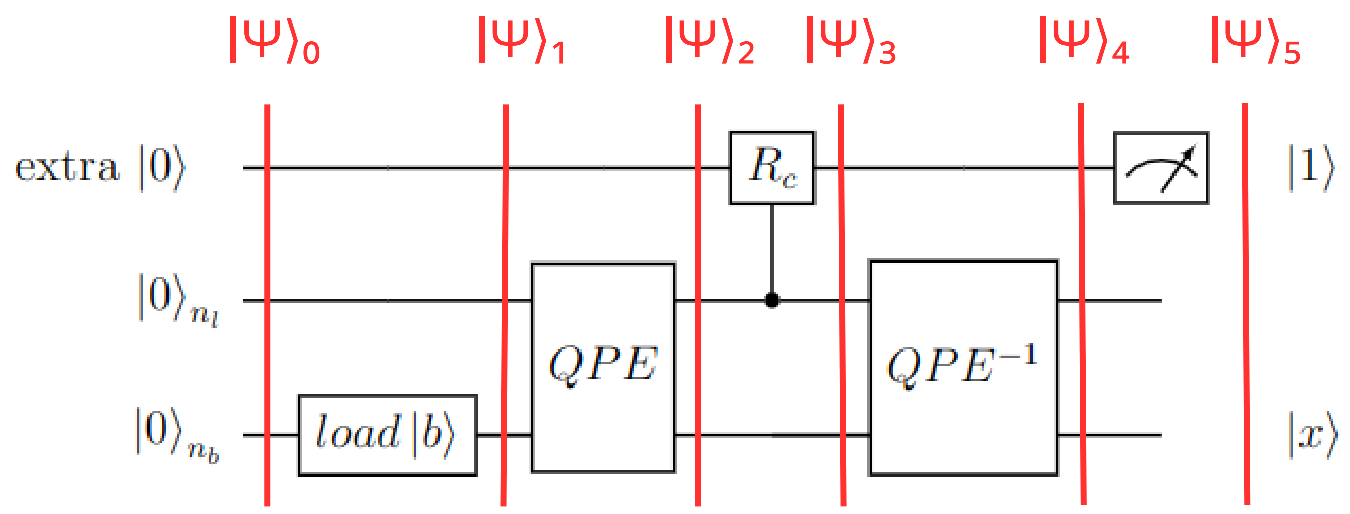


Fig 23: Circuit of the HHL algorithm

$$|\psi_4\rangle \longmapsto |\psi_5\rangle = \left(\sqrt{\frac{1}{\sum_i C^2|b_i|^2/|\lambda_i|^2}}\right)\sum_i C\frac{b_i}{\lambda_i}\,|u_i\rangle_{n_b}\,|0\rangle_{n_l}$$

Fig 29: The extra qubit is measured

VI Harrow-Hassidim-Lloyd algorithm:
Time complexity

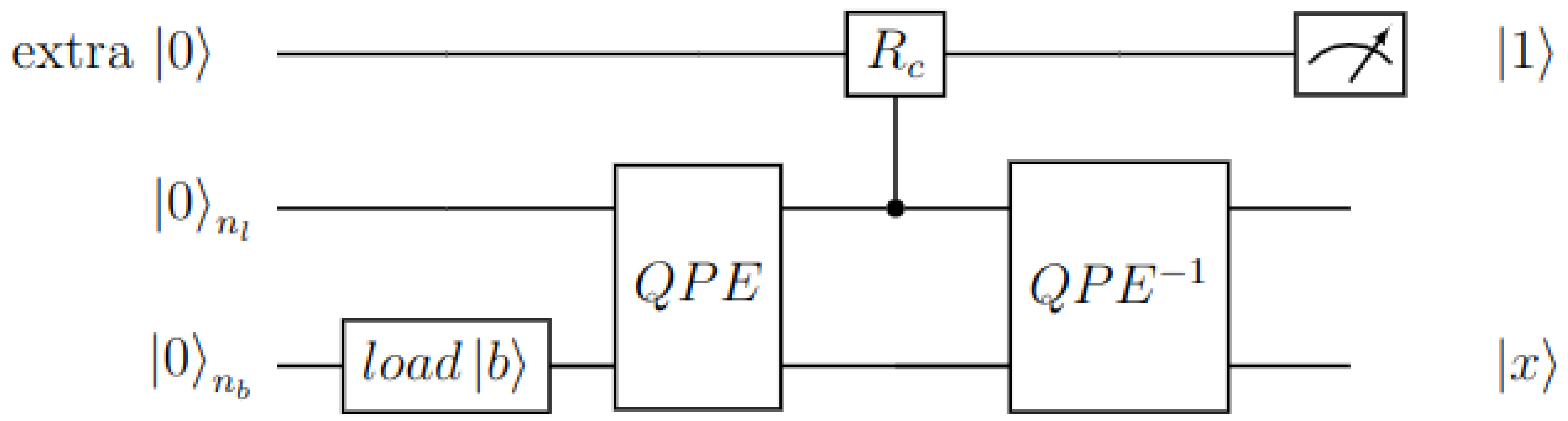


Fig 23: Circuit of the HHL algorithm

The cost of the HHL is $O(\log(N)) = O(n)$

VI Harrow-Hassidim-Lloyd algorithm:
Back to the example

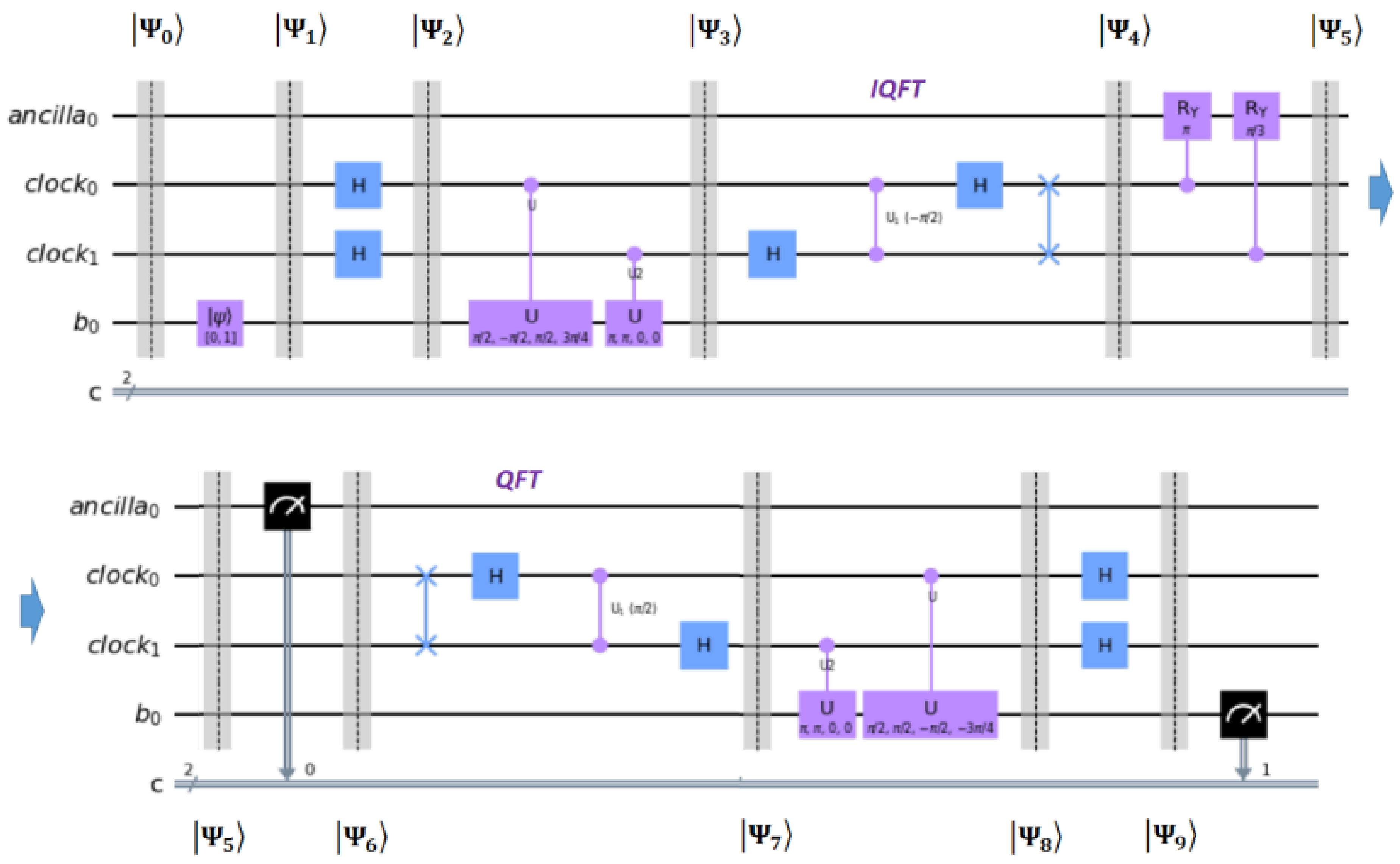


Fig 30: Circuit that solves the QLSP example[3]

VI Harrow-Hassidim-Lloyd algorithm:
Back to the example

Example:

$$A = \begin{pmatrix} 1 & -\frac{1}{3} \\ -\frac{1}{3} & 1 \end{pmatrix}$$

$$\vec{b} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$|b\rangle = |1\rangle$

$$\vec{x} = \begin{pmatrix} \frac{3}{8} \\ \frac{9}{8} \end{pmatrix}$$

$|x_0|^2 : |x_1|^2 = 1 : 9$

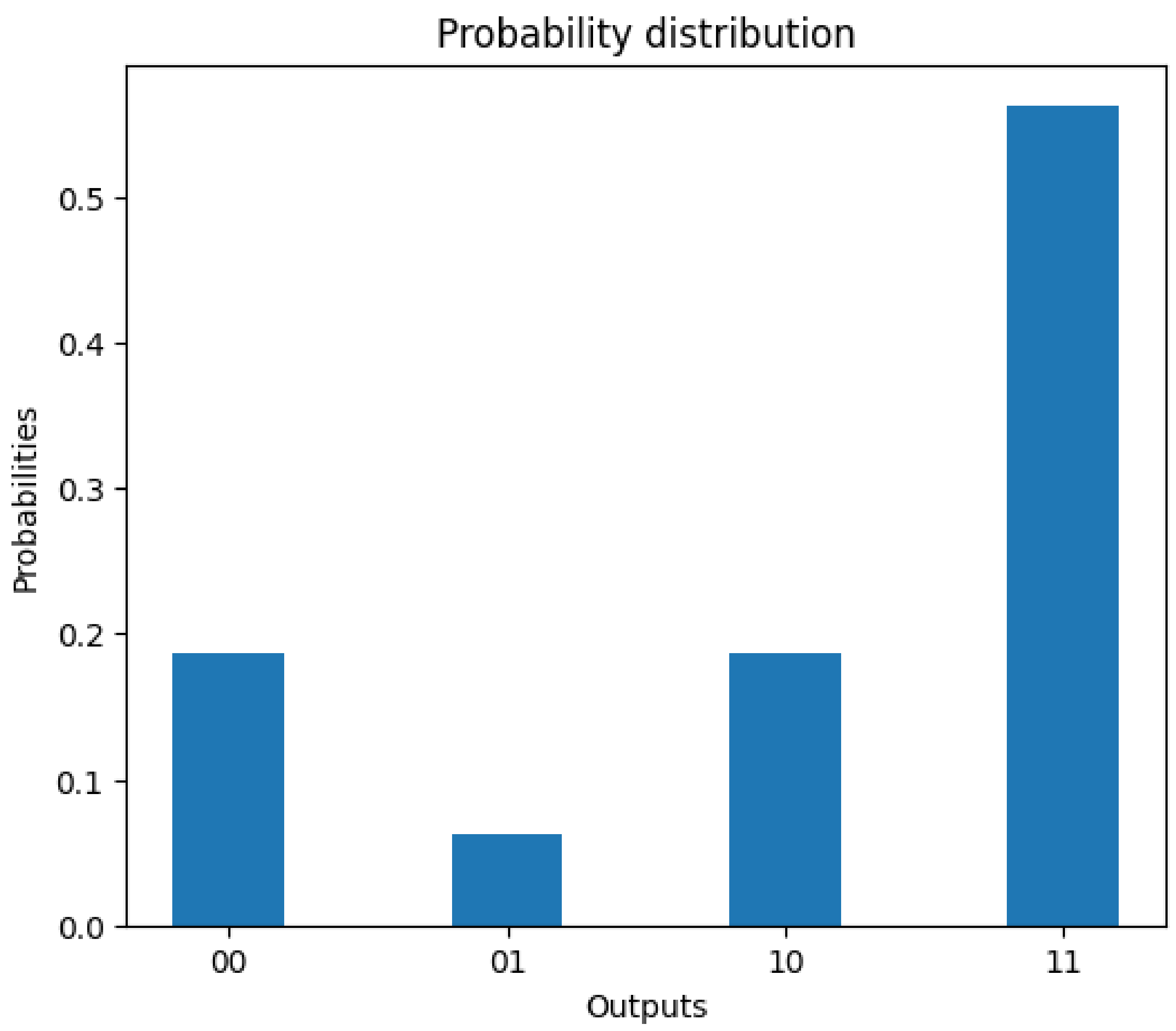


Fig 31: probability distribution of the resulting state

Ratio of the resulting state:

0.0625 : 0.5625 = 1 : 9

VII Qibo

Open source framework for quantum computing

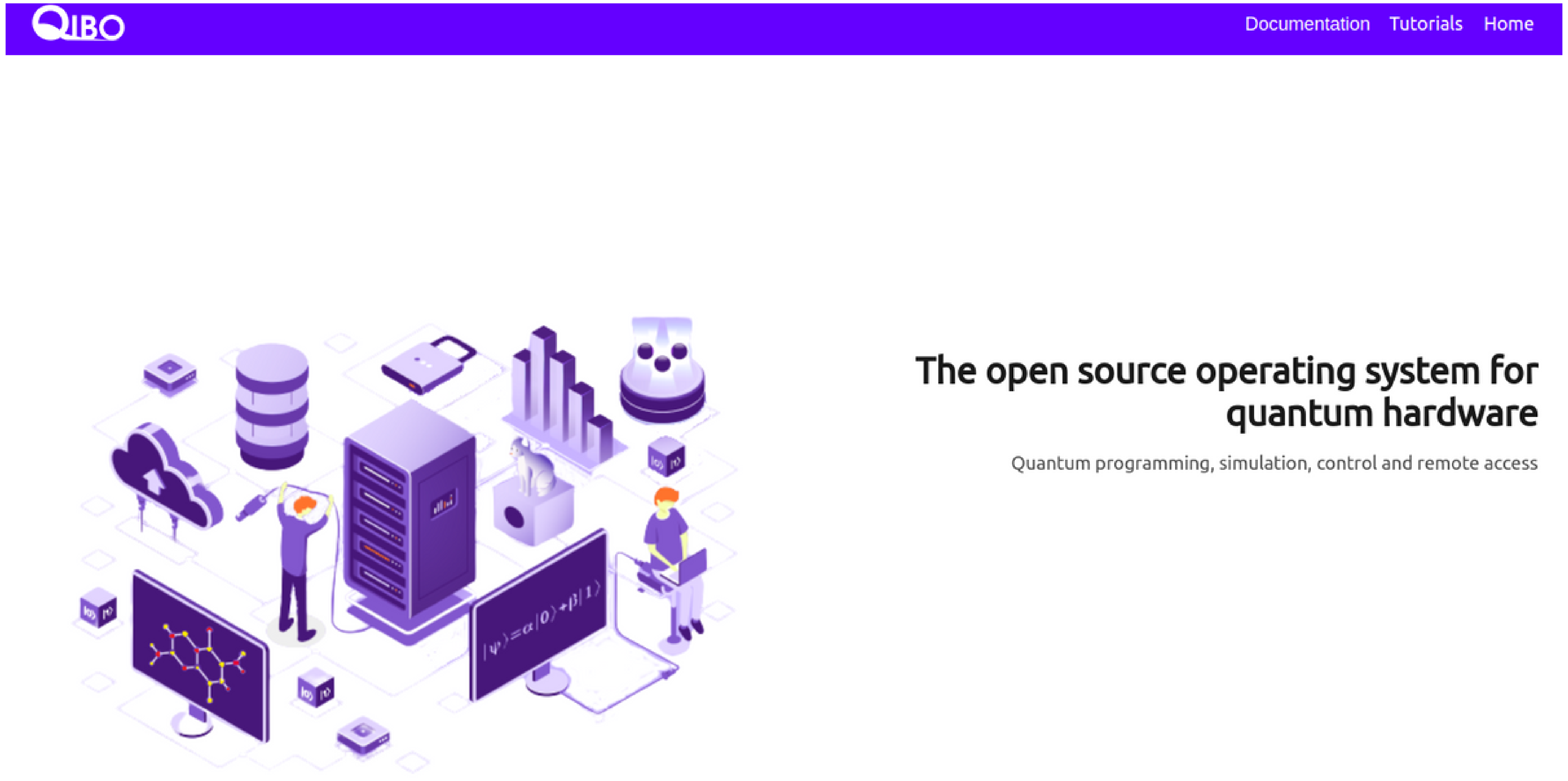


Fig 32: Official Qibo's website [4]

```python
import numpy as np
from qibo.models import Circuit
from qibo import gates
from qibo.config import raise_error
from qibo.models.circuit import Circuit
import math

nqubits=7
circuit = Circuit(nqubits)
for q1 in range(nqubits):
    circuit.add(gates.H(q1))
    for q2 in range(q1 + 1, nqubits):
        theta = math.pi / 2 ** (q2 - q1)
        circuit.add(gates.CU1(q2, q1, theta))

for i in range(nqubits // 2):
    circuit.add(gates.SWAP(i, nqubits - i - 1))

print(circuit.draw())
```

Fig 33: QFT in Qibo

VIII Conclusion and future work

Contribution to Qibo

This project will be able to contribute to the development of the open source framework.

Impact of Quantum Computing

Quantum computing can bring a big impact in society and computer scientist have to be part of it.

Study of the general implementation

The implementation of the hamiltonian, the rotation and the state loader.

Study of applications

Applications in quantum machine learning.

IX References

1. Harrow, Aram W., et al. Quantum algorithm for solving linear systems of equations. Physical Review Letters, vol. 103, n.o 15, octubre de 2009, p. 150502. arXiv.org, https://doi.org/10.1103/PhysRevLett.103.150502
2. Di Adamo, Stephen, et al. «Distributed Quantum Computing and Network Control for Accelerated VQE». IEEE Transactions on Quantum Engineering, vol. 2, 2021, pp. 1-21. DOI.org (Crossref), https://doi.org/10.1109/TQE.2021.3057908
3. Morrell Jr, Hector Jose, et al. Step-by-Step HHL Algorithm Walkthrough to Enhance the Understanding of Critical Quantum Computing Concepts. arXiv. arXiv.org, http://arxiv.org/abs/2108.09004.
4. website of the Qibo framework, https://qibo.science

All references of this project can be found in the thesis report in https://github.com/alexao8/TFG-HHL_algorithm/tree/main

26/06/2023

School of informatics of
Barcelona - UPC

SUPERVISORS:
JORDI DELGADO PIN (UPC)
ALBA CERVERA LIERTA (BSC-CNS)

ALEXANDRE ALEMANY
ORFILA

Harrow-Hassidim-Lloyd algorithm:
Study and implementation using Qibo

FINAL PROJECT DEFENCE