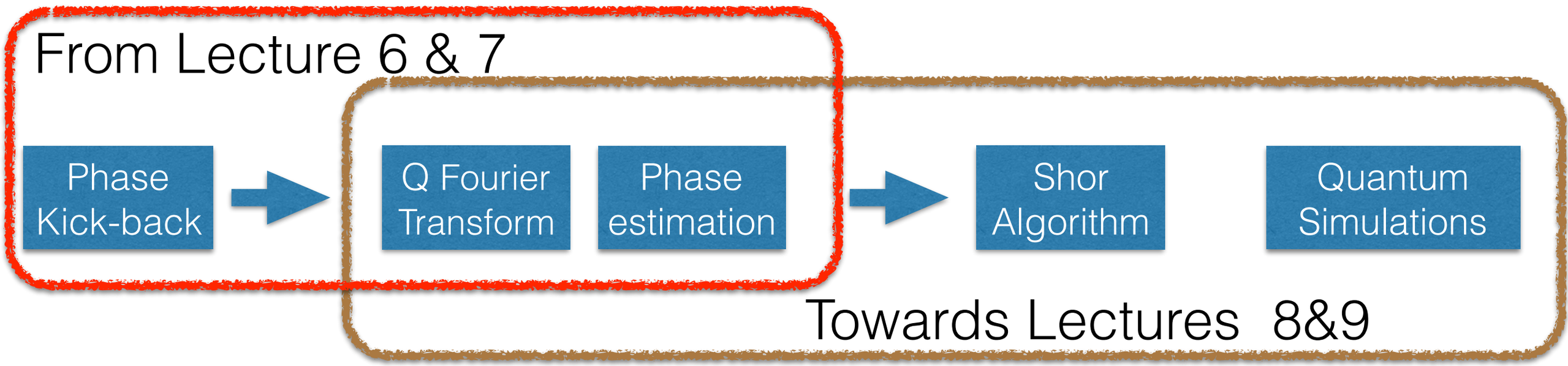
Quantum Information Processing

Budapest University of Technology and Economics
2019 Spring

Ákos Budai, András Pályi, Zoltán Zimborás

Lecture 8 (Quantum Algorithms 4): Shor's Algorithm



Shor’s algorithm in short

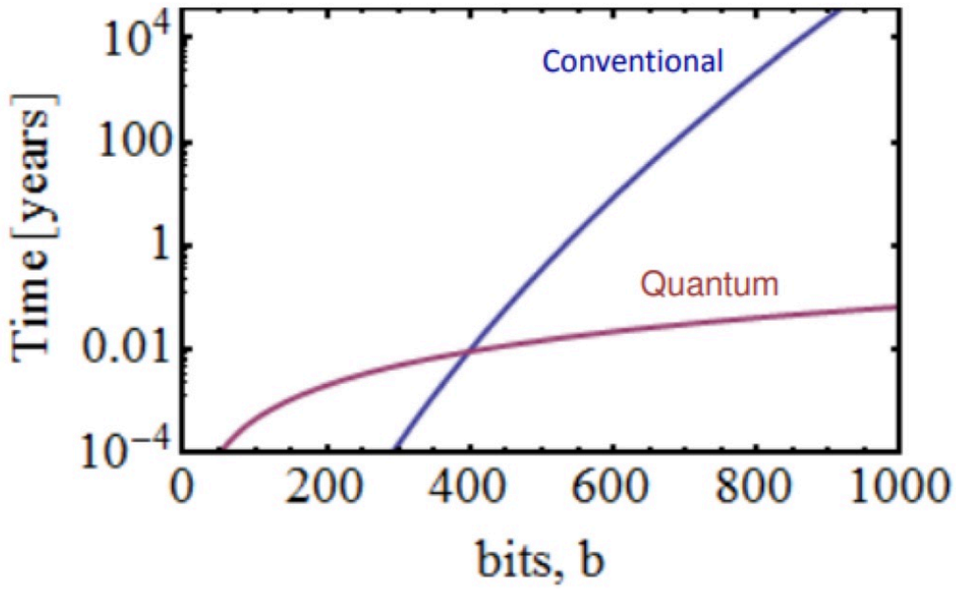
Peter Shor

Shor's algorithm is a quantum algorithm for factoring a number N in $O(n^3)$ time, named after Peter Shor.

Factor a number into primes:

M = p * q

How long will it take ? (t)



Source: http://www.ibm.com/

<u>Classical</u>

$t \sim O(2^{n^{(1/2)}})$

<u>Quantum</u>

$t \sim O(n^3)$

$$\gcd(a^{r/2} - 1, N) \quad \gcd(a^{r/2} + 1, N)$$

An algorithm to factor numbers

1. Pick a random number $a < N$.

2. Compute **gcd**(a,N), the greatest common divisor of a and N.

3. If **gcd**$(a,N)\neq 1$, then this number is a nontrivial factor of N, so we are done.

4. Otherwise, use a period-finding subroutine to find r which denotes the period of the following function:

$$f(x) = a^x \bmod N$$

5. If r is odd, or a to the power of $r/2$ gives $N-1$ modulo N, then go back to step 1.

6. Otherwise, we have a nontrivial factor of N:

$$\gcd(a^{r/2} - 1, N) \quad \gcd(a^{r/2} + 1, N)$$

Modular multiplication and period finding

$$
\begin{array}{ccccccccccc}
|1\rangle & \rightarrow & |7\rangle & \rightarrow & |4\rangle & \rightarrow & |13\rangle & \rightarrow & |1\rangle \\
|2\rangle & \rightarrow & |14\rangle & \rightarrow & |8\rangle & \rightarrow & |11\rangle & \rightarrow & |2\rangle
\end{array}
$$

Multiplication by 7 modulo 15

$$
\begin{aligned}
7^2 &= 4 \pmod{15} \\
7^3 &= 4 \cdot 7 = 13 \pmod{15} \\
7^4 &= 13 \cdot 7 = 1 \pmod{15}
\end{aligned}
$$

$\gcd(a^{r/2} - 1, N)$ $\gcd(a^{r/2} + 1, N)$

Recap: Quantum Fourier Transform

Applying the *quantum* Fourier transform means that we do a DFT on the *amplitudes* of a quantum state:

$$\sum_j \alpha_j |j\rangle \rightarrow \sum_k \tilde{\alpha}_k |k\rangle, \quad \text{where} \quad \tilde{\alpha}_k \equiv \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} e^{2\pi i j k/N} \alpha_j.$$

n qubit computational qubit basis:

$$|x_1\rangle \otimes |x_2\rangle \otimes |x_3\rangle \otimes \cdots \otimes |x_n\rangle$$

$|x\rangle \quad x \in \{0,1\}^n$ n bit string

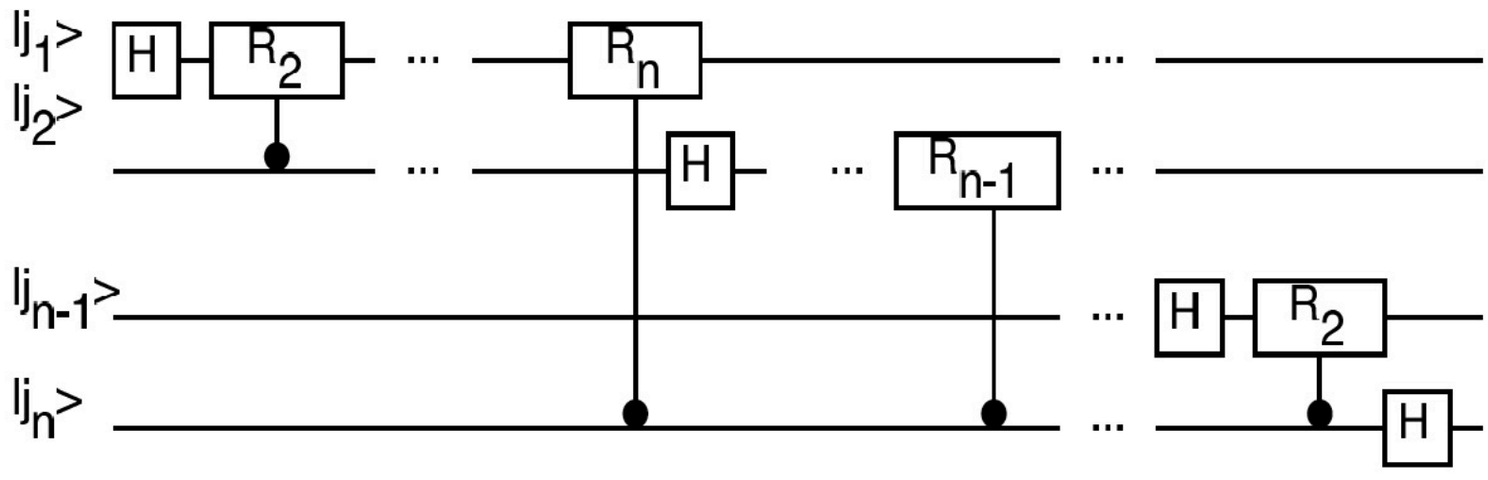
$$|x\rangle \quad x \in \{0, 1, 2, \ldots, 2^n - 1\}$$

Recap: Circuits for the Quantum Fourier Transform

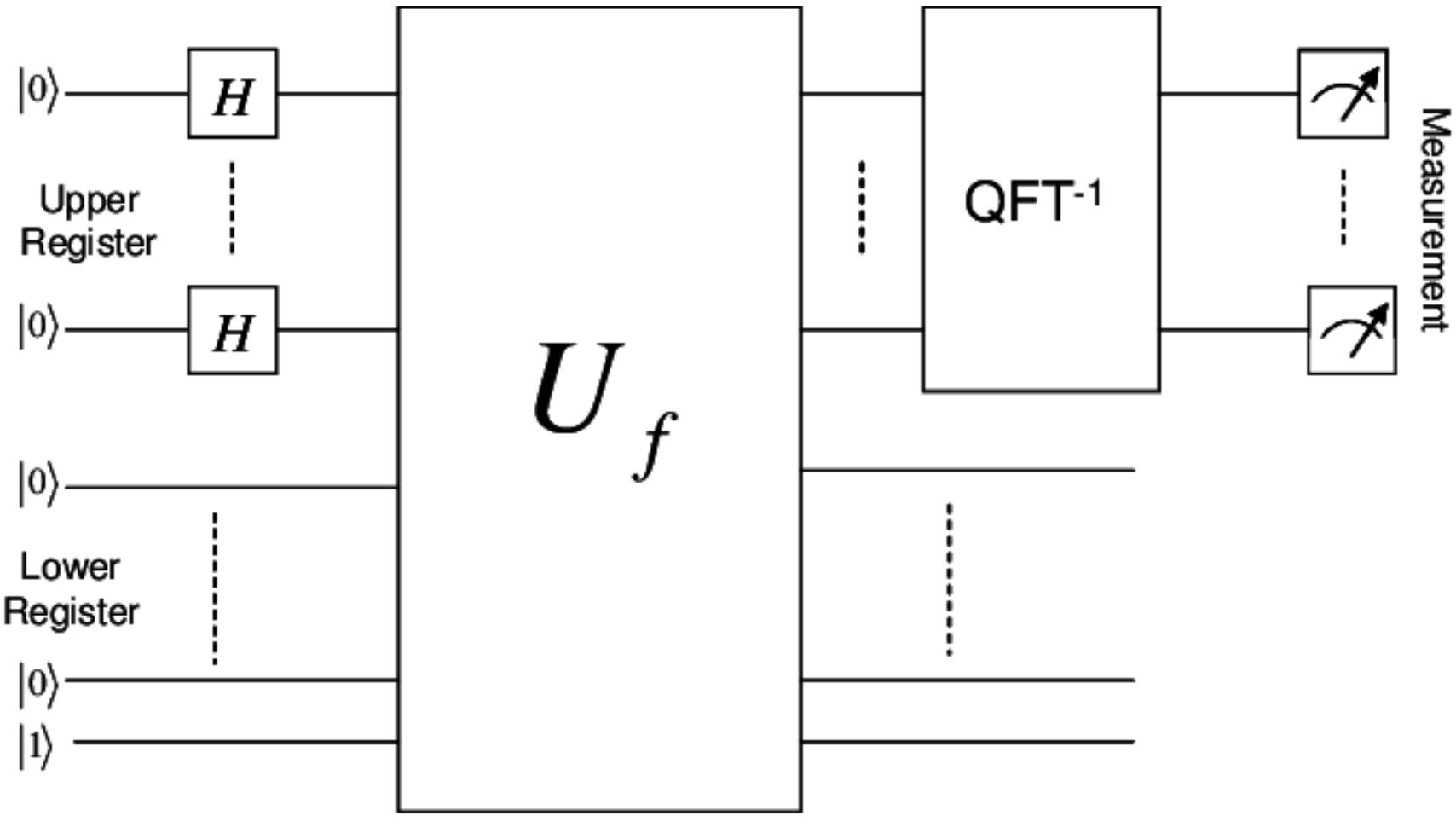
Define the rotation

$$\hat{R}_k = \begin{pmatrix} 1 & 0 \\ 0 & e^{2\pi i/2^k} \end{pmatrix}.$$

The controlled-$\hat{R}_k$ gate performs this if and only if a control bit is $|1\rangle$ rather than $|0\rangle$. Putting these together with the Hadamards gives the following circuit:



Shor's algorithm as a black box algorithm



Shor's algorithm as a black box algorithm

1. *Initialization*

$$Q^{-1/2}\sum_{x=0}^{Q-1}|x\rangle|0\rangle$$

a superposition of Q states

2. *Construction*

$$Q^{-1/2}\sum_{x=0}^{Q-1}|x\rangle|f(x)\rangle$$

f(x) as a quantum function and apply it to the above state

3. *Transformation*

$$Q^{-1}\sum_{x}\sum_{y}\omega^{xy}|y\rangle|f(x)\rangle$$

apply the quantum Fourier transform, and leads to the final state

Shor's algorithm as a black box algorithm

1.*Initialization*

$$Q^{-1/2}\sum_{x=0}^{Q-1}|x\rangle|0\rangle$$

a superposition of Q states

2.*Construction*

$$Q^{-1/2}\sum_{x=0}^{Q-1}|x\rangle|f(x)\rangle$$

f(x) as a quantum function and apply it to the above state

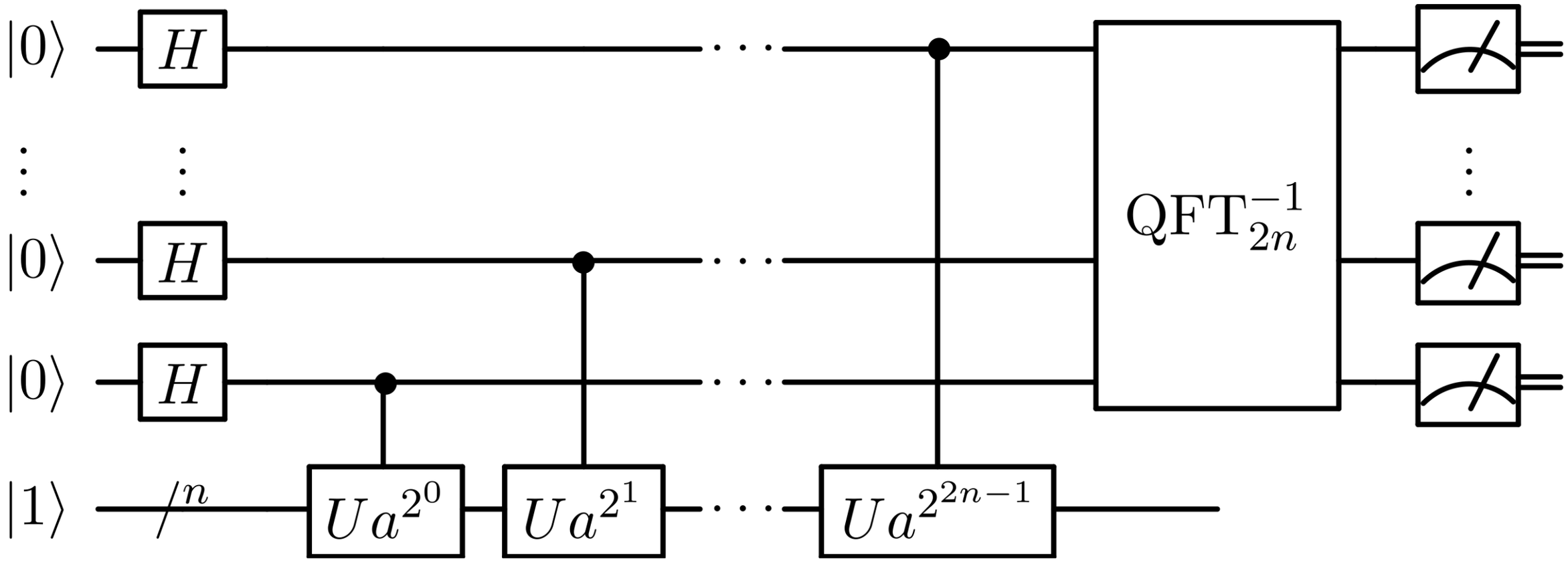
3.*Transformation*

$$Q^{-1}\sum_{x}\sum_{y}\omega^{xy}|y\rangle|f(x)\rangle$$

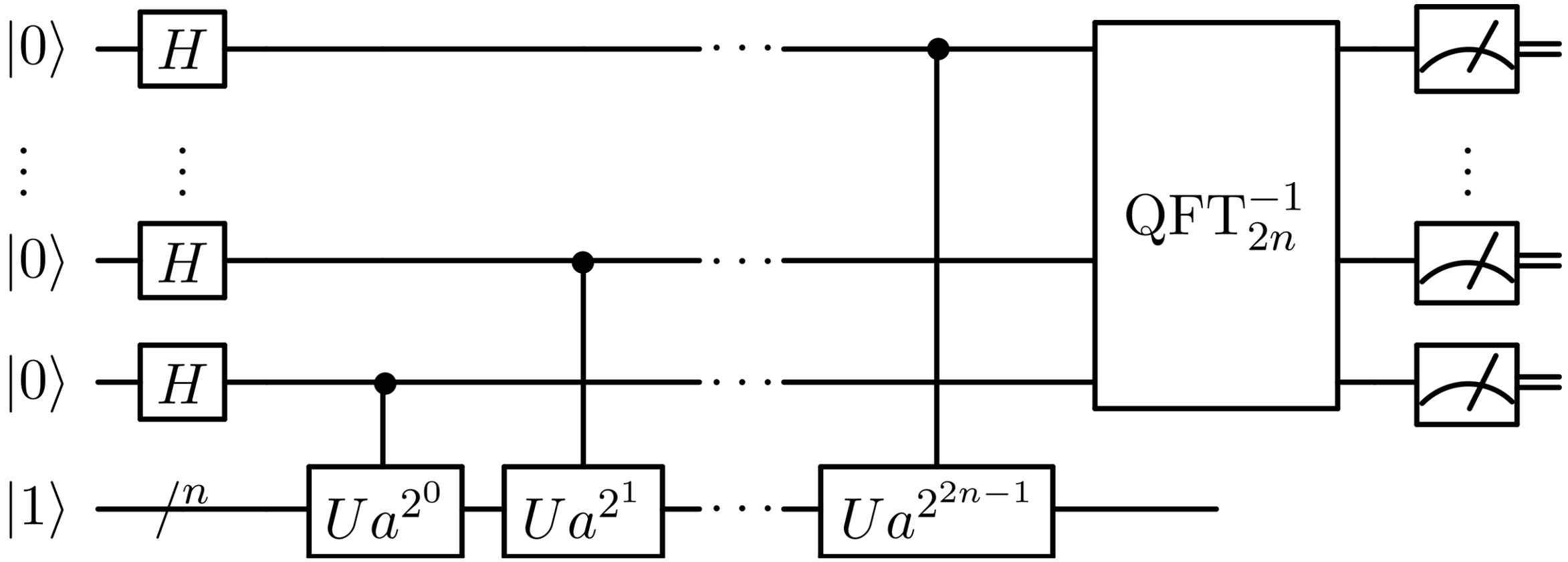
apply the quantum Fourier transform, and leads to the final state

$$P\left(\frac{y}{2^Q}\neq\frac{m}{r}\right)\leq\frac{1}{Q}$$

Decomposing the Black Box in Shor's algorithm

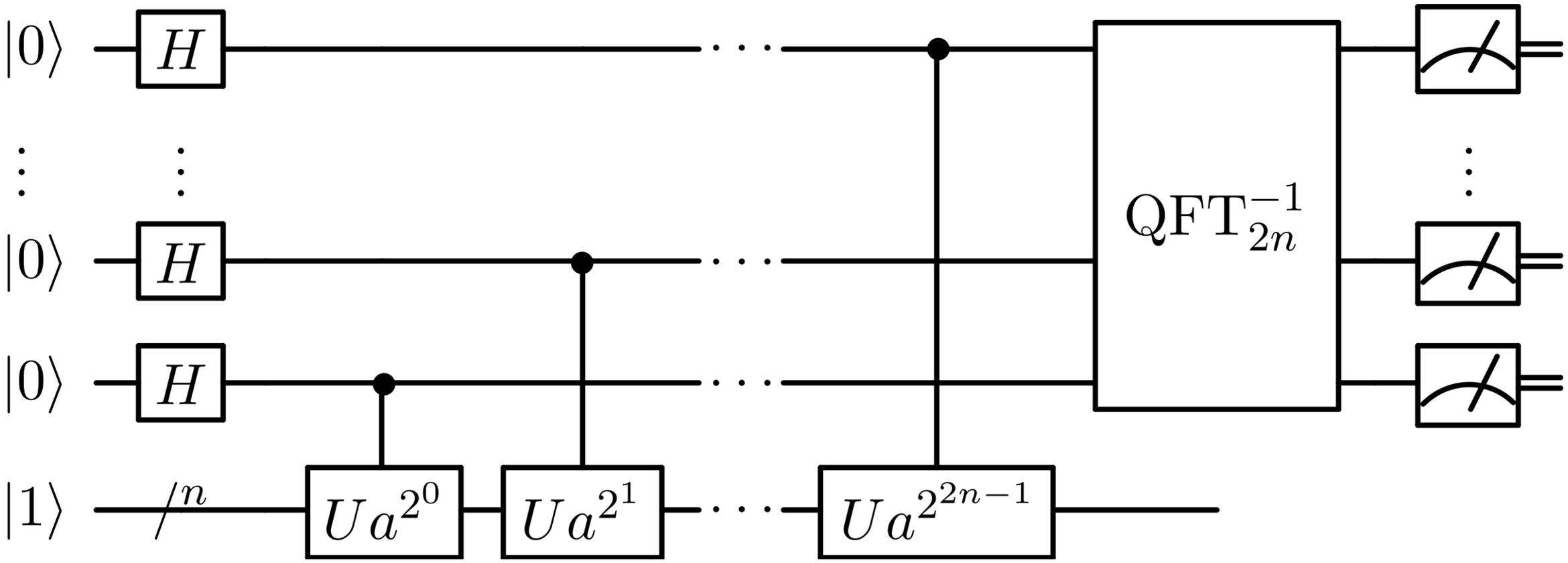


Decomposing the Black Box in Shor's algorithm



If one has an efficient description of a unitary in terms of a circuit, it is easy to make an efficient circuit description of the controlled version of this unitary:

Decomposing the Black Box in Shor's algorithm

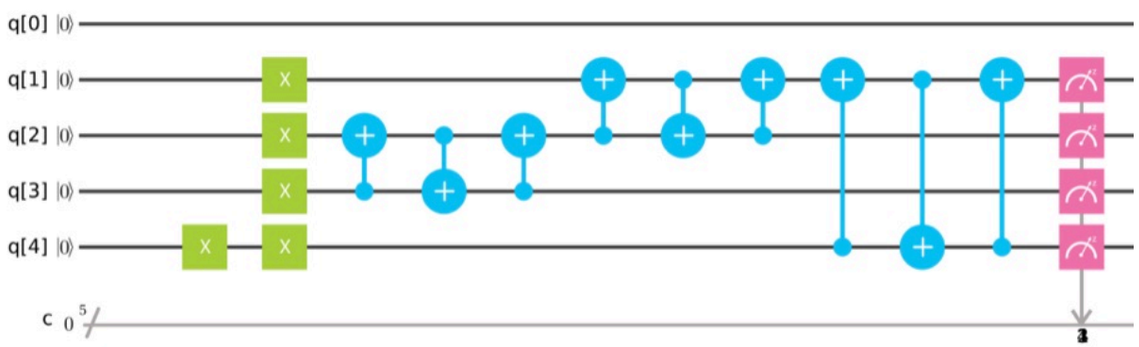


If one has an efficient description of a unitary in terms of a circuit, it is easy to make an efficient circuit description of the controlled version of this unitary: each elementary gate in the circuit has to be replaced by its controlled version

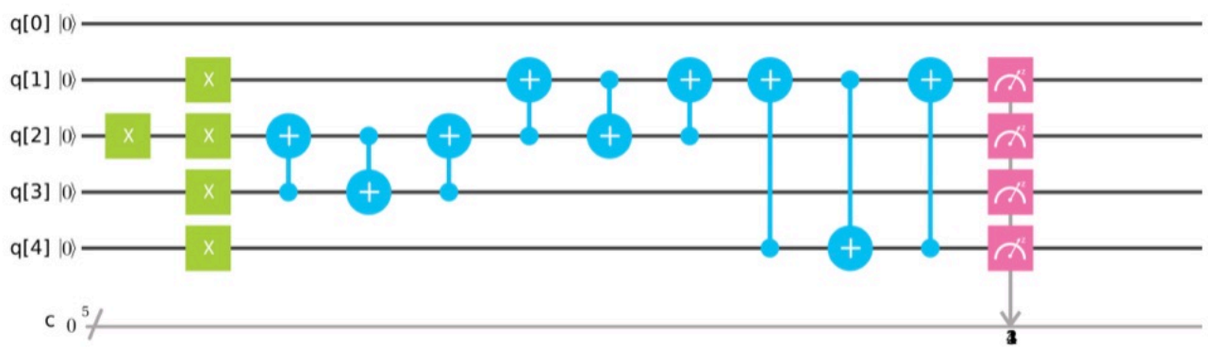
Decomposing the Black Box in Shor's algorithm

There exists efficient circuit descriptions of the modular multiplication

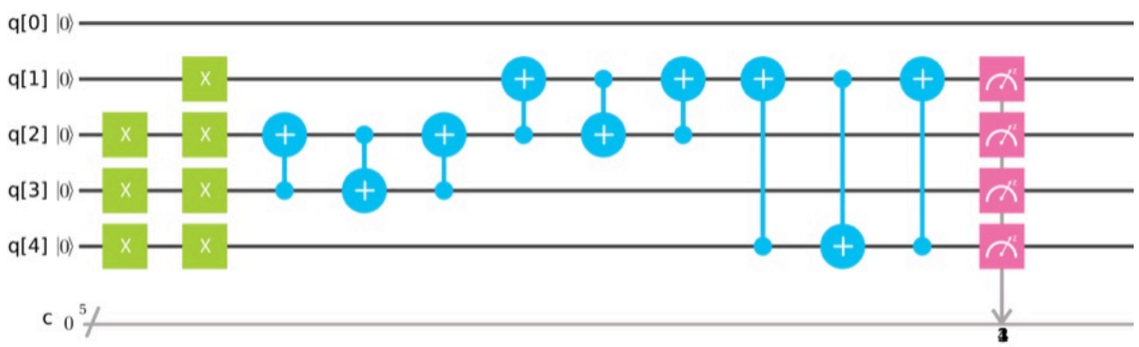
Multi7x1Mod15



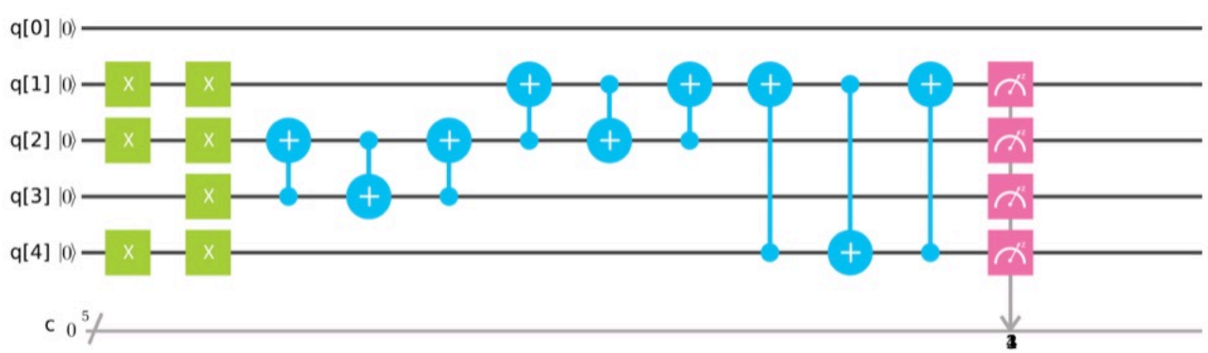
Multi7x4Mod15



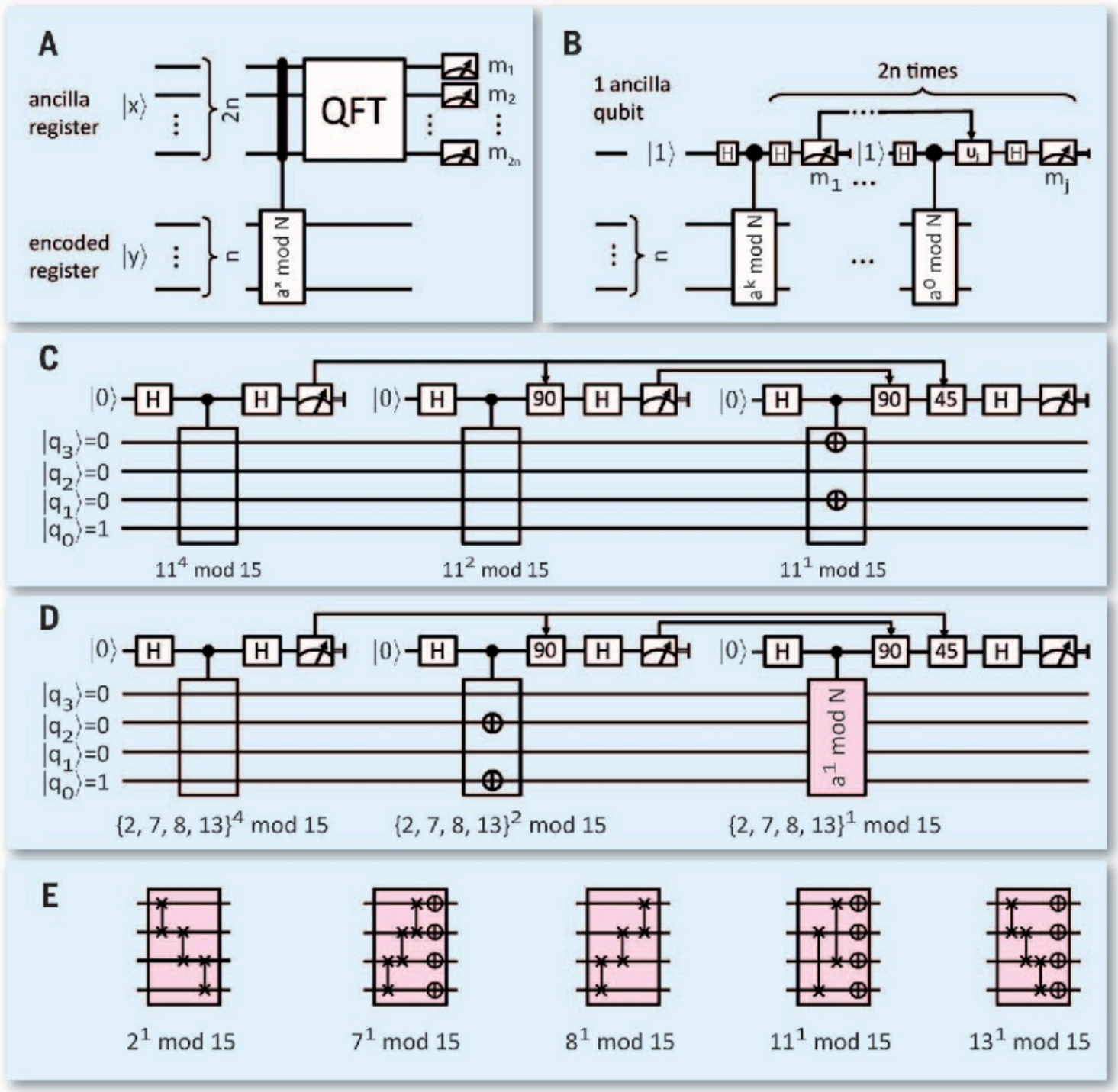
Multi7x7Mod15



Multi7x13Mod15



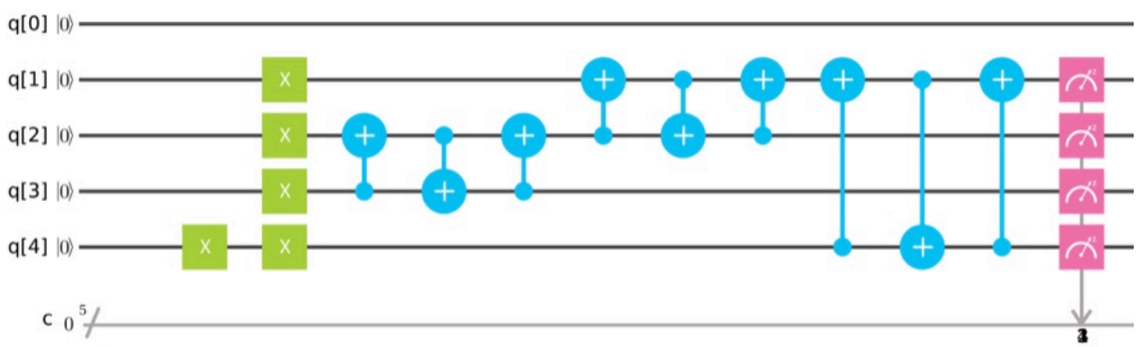
Optimizing Shor's algorithm



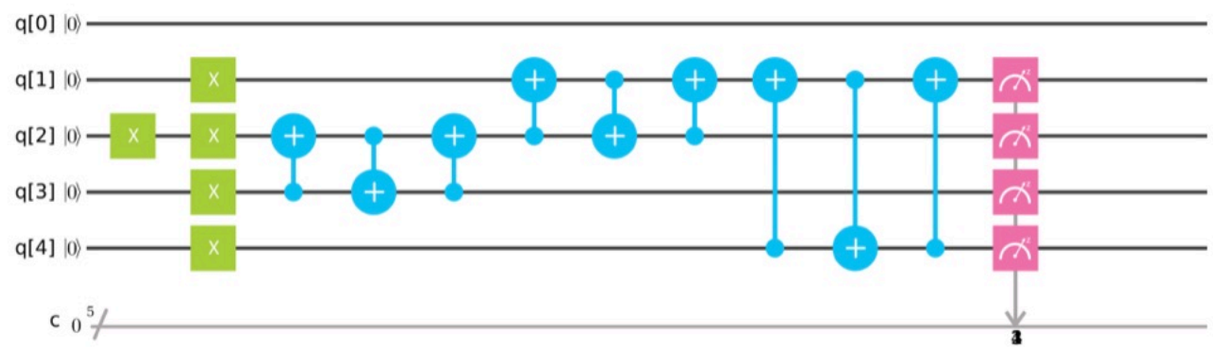
Exercise 1

Build the circuit below with Qiskit, and show that it implements the modular multiplication

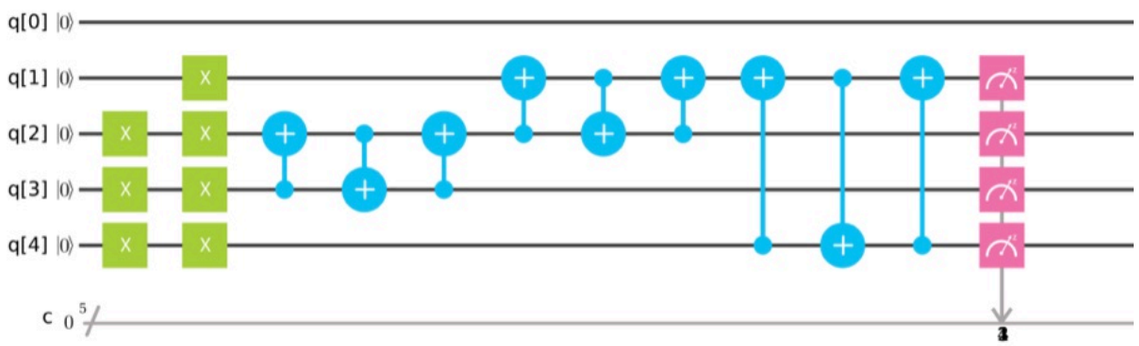
Multi7x1Mod15



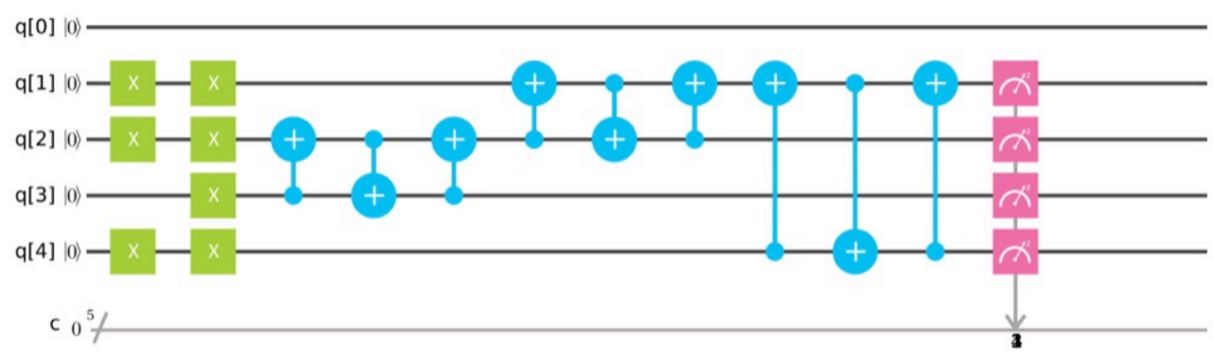
Multi7x4Mod15



Multi7x7Mod15



Multi7x13Mod15



Exercise 2

Using the notebook at
github.com/Qiskit/qiskit-tutorials/blob/master/community/algorithms/shor_algorithm.ipynb
implant the period finding part of Shor's algorithm with an optimized circuit