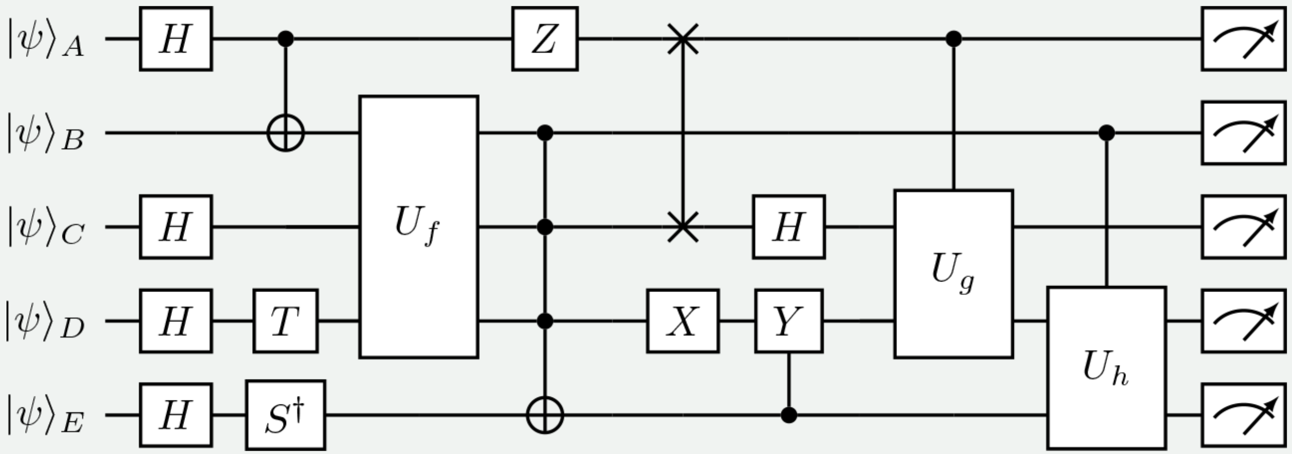
Operating with Quantum Integers:

an Efficient ‘Multiples of’ Oracle

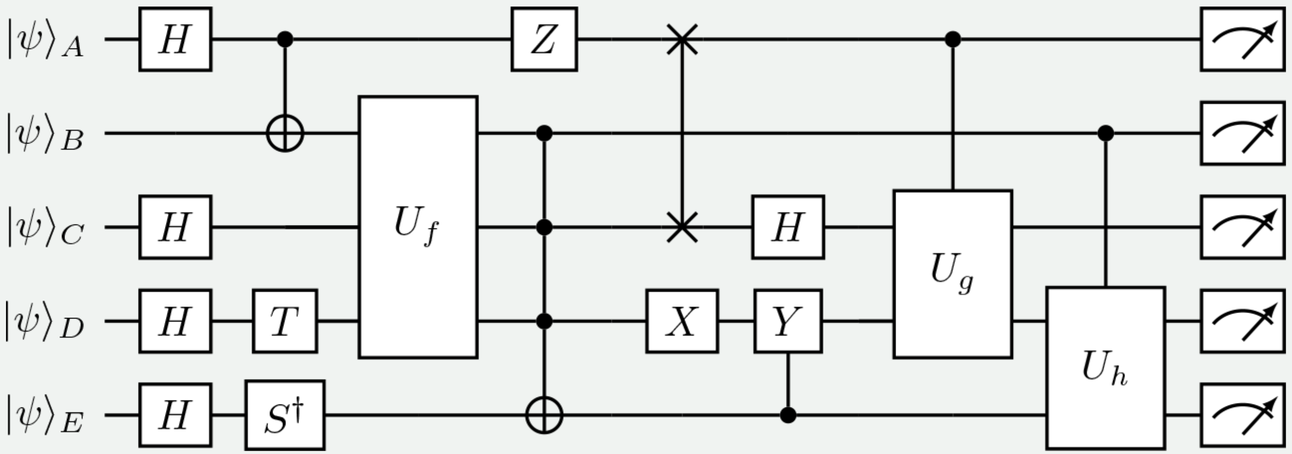
Javier Sánchez Rivero, Daniel Talaván,
José García Alonso, Antonio Ruiz Cortés, Juan Manuel Murillo

Quantum Computing



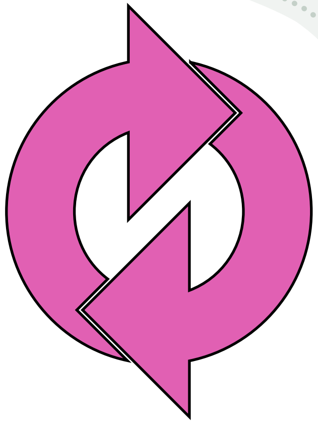
Software Engineering Tools

Other fields
Physicists
Mathematicians

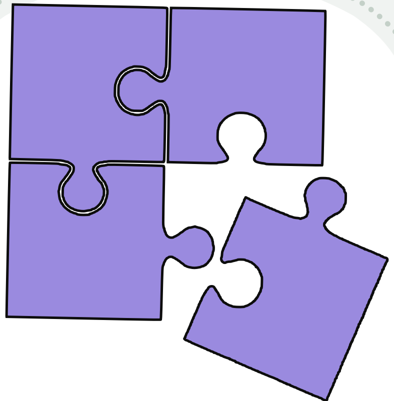


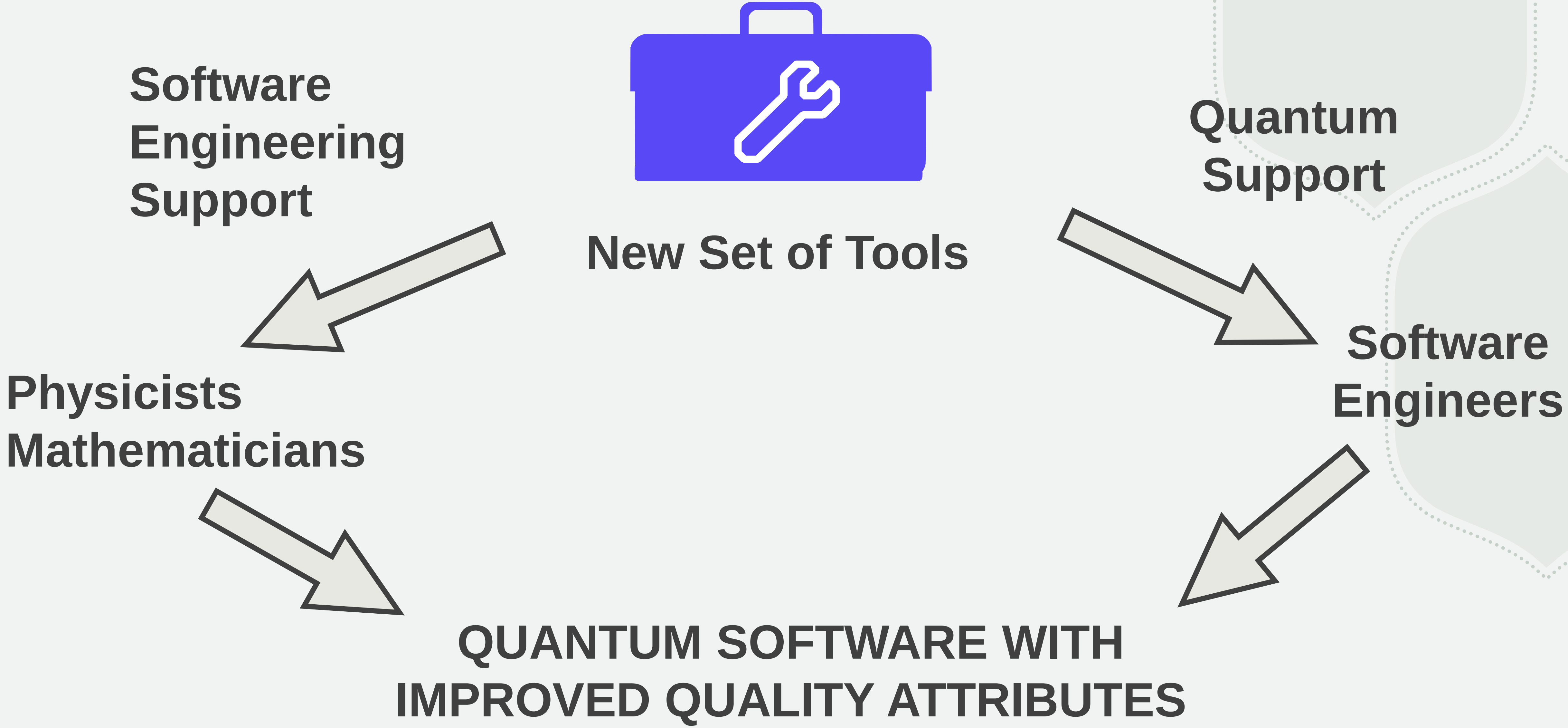
Quality

Reusability



Composability




Software Engineering Support
New Set of Tools
Quantum Support
Physicists
Mathematicians
Software Engineers
QUANTUM SOFTWARE WITH IMPROVED QUALITY ATTRIBUTES

Outline

- Grover’s Algorithm and Amplitude Amplification
- Implementation of Grover’s Algorithm
- ‘Multiples of’ Oracle
- Algorithm for Less-Than Oracle
- Discussion
- Summary and Future Works

Grover’s Algorithm

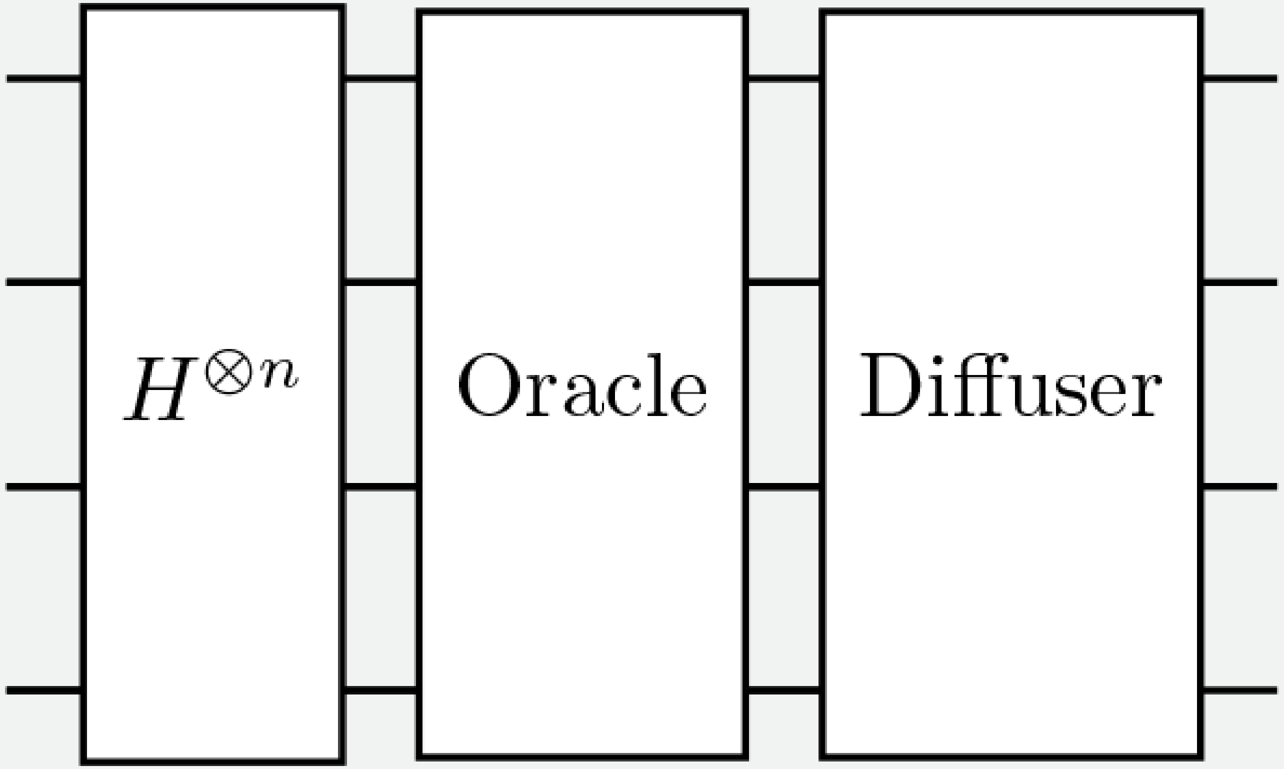
- Quantum searching algorithm.
- n qubits, $N = 2^n$ states.
- $\mathcal{O}(\sqrt{N})$

- Amplitude Amplification searches for multiple values, M.
- $\mathcal{O}(\sqrt{N/M})$

Lov K. Grover

Principles of Quantum Artificial Intelligence: Quantum Problem Solving and Machine Learning, 2nd Edition - Scientific Figure on ResearchGate. Available from: https://www.researchgate.net/figure/Lov-Kumar-Grover-originator-of-the-Grovers-search-algorithm_fig4_342012452

Grover's Algorithm Implementation

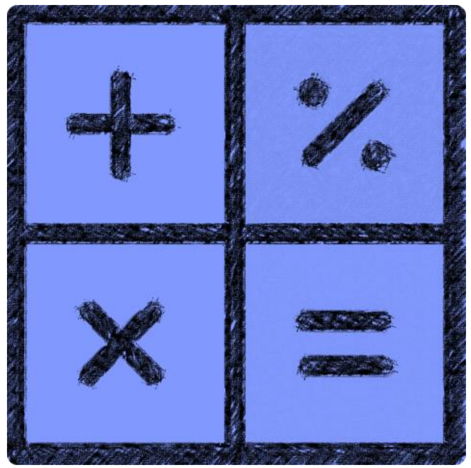


Oracle ‘Multiples of’

- Quantum states as natural numbers

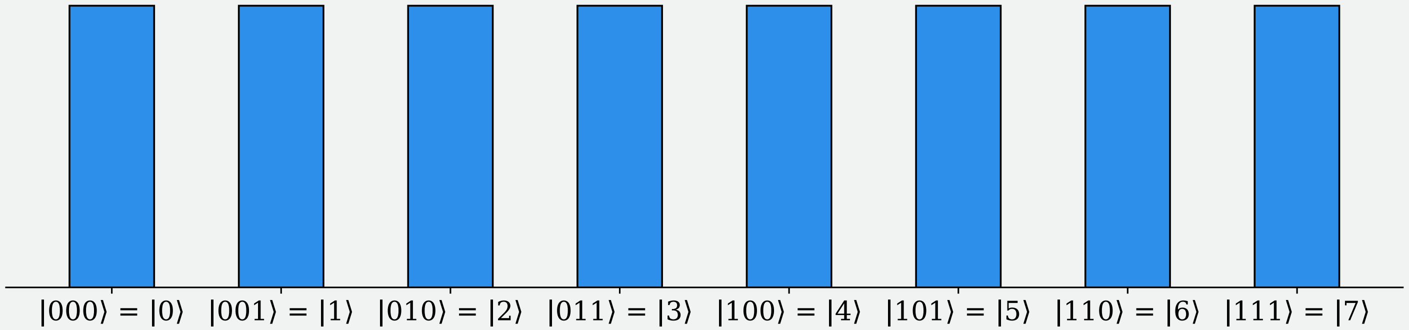
$$|101\rangle = |5\rangle$$
$$|1001\rangle = |9\rangle$$
$$|0011\rangle = |3\rangle$$

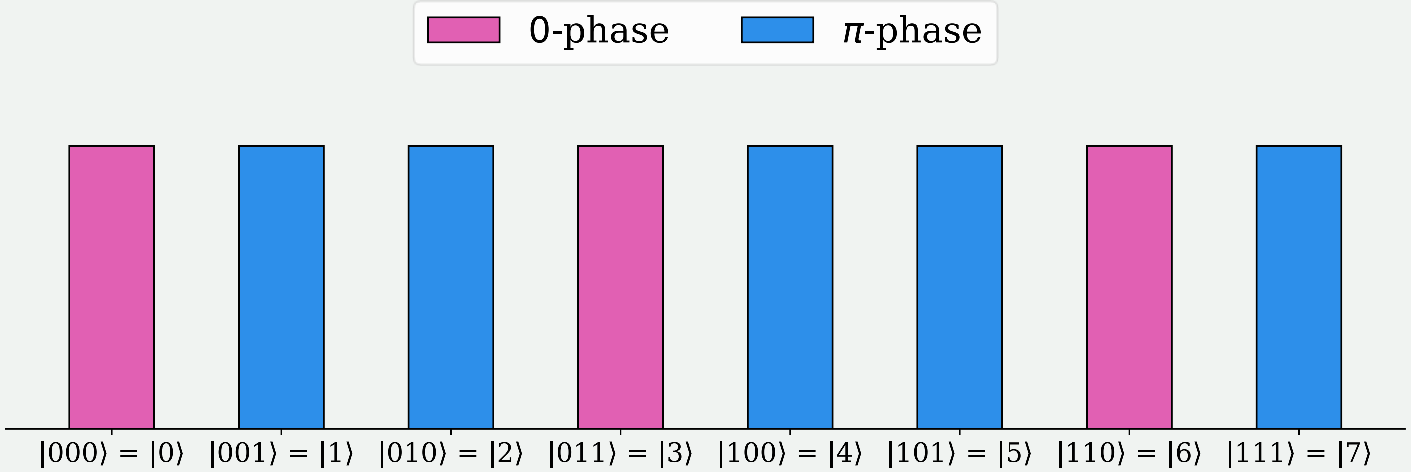
- Phase oracle for Amplitude Amplification
- Given number $k \in \mathbb{N}$
- π-phase to numbers multiples of k



Oracle

‘Multiples of 3’





Algorithm for generating the Oracle

- Classical algorithm to build the ‘Multiples of’ Oracle.
- Input:
  - Number of qubits: n
  - $k \in \mathbb{N}$
- Output:
  - Quantum circuit which implements the oracle.

Al-Khwarizmi

Idea inspiring the algorithm

$$M \in \mathbb{N}, \qquad M = \sum_{i=0}^{m} a_i \cdot 2^i$$

Idea inspiring the algorithm

$$M \in \mathbb{N}, \qquad M = \sum_{i=0}^{m} a_i \cdot 2^i$$

$$23 = 10111_2 = 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0$$

Idea inspiring the algorithm

$$M \in \mathbb{N}, \qquad M = \sum_{i=0}^{m} a_i \cdot 2^i$$

$$23 = 10111_2 = 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0$$

Is 23 multiple of 5? $\quad 23 \equiv 3 \bmod 5$

$$23 = 10111_2$$

Idea inspiring the algorithm

$$r_i \equiv 2^i \, mod \, k, \; 0 \leq r_i < k$$

Idea inspiring the algorithm

$$r_i \equiv 2^i \; mod \; k, \; 0 \leq r_i < k$$

Power of 2	Remainder of $2^{\mathrm{i}}, k = 5$
$2^0 = 1$	$r_0 = 1$
$2^1 = 2$	$r_1 = 2$
$2^2 = 4$	$r_2 = 4$
$2^3 = 8$	$r_3 = 3$
$2^4 = 16$	$r_4 = 1$
$2^5 = 32$	$r_5 = 2$
$2^6 = 64$	$r_6 = 4$

Idea inspiring the algorithm

$$r_i \equiv 2^i \bmod k,\ 0 \le r_i < k$$

Power of 2	Remainder of $2^{\mathrm{i}}, k = 5$
$2^0 = 1$	$r_0 = 1$
$2^1 = 2$	$r_1 = 2$
$2^2 = 4$	$r_2 = 4$
$2^3 = 8$	$r_3 = 3$
$2^4 = 16$	$r_4 = 1$
$2^5 = 32$	$r_5 = 2$
$2^6 = 64$	$r_6 = 4$

For $23 = 10111_2$

$$\begin{aligned} 23 &\equiv 1 \cdot r_4 + 0 \cdot r_3 + 1 \cdot r_2 + 1 \cdot r_1 + 1 \cdot r_0 \\ &\equiv 1 \cdot 1 + 0 \cdot 3 + 1 \cdot 4 + 1 \cdot 2 + 1 \cdot 1 \\ &\equiv 1 + 4 + 2 + 1 \\ &\equiv 8 \equiv 3 \bmod 5 \end{aligned}$$

Idea inspiring the algorithm

$$r_i \equiv 2^i \; mod \; k, \; 0 \le r_i < k$$

Power of 2	Remainder of $2^{\mathrm{i}}, k = 5$
$2^0 = 1$	$r_0 = 1$
$2^1 = 2$	$r_1 = 2$
$2^2 = 4$	$r_2 = 4$
$2^3 = 8$	$r_3 = 3$
$2^4 = 16$	$r_4 = 1$
$2^5 = 32$	$r_5 = 2$
$2^6 = 64$	$r_6 = 4$

For $23 = 10111_2$

$$23 \equiv 3 \; mod \; 5$$

For $25 = 11001_2$

$$\begin{aligned} 25 &\equiv 1 \cdot r_4 + 1 \cdot r_3 + 0 \cdot r_2 + 0 \cdot r_1 + 1 \cdot r_0 \\ &\equiv 1 \cdot 1 + 1 \cdot 3 + 0 \cdot 4 + 0 \cdot 2 + 1 \cdot 1 \\ &\equiv 1 + 3 + 1 \\ &\equiv 5 \equiv 0 \; mod \; 5 \end{aligned}$$

Idea inspiring the algorithm

$$r_i \equiv 2^i \, mod \, k, \; 0 \leq r_i < k$$

Power of 2	Remainder of $2^{\mathrm{i}}, k = 5$
$2^0 = 1$	$r_0 = 1$
$2^1 = 2$	$r_1 = 2$
$2^2 = 4$	$r_2 = 4$
$2^3 = 8$	$r_3 = 3$
$2^4 = 16$	$r_4 = 1$
$2^5 = 32$	$r_5 = 2$
$2^6 = 64$	$r_6 = 4$

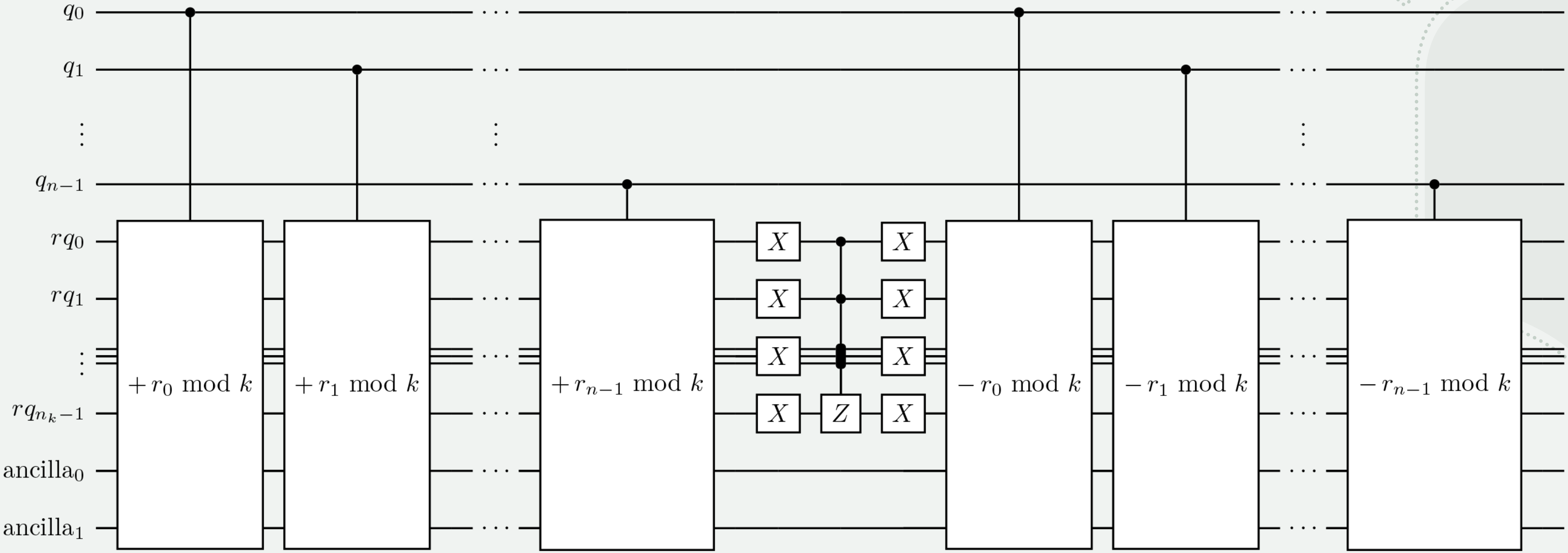
For $23 = 10111_2$

$$23 \equiv 3 \, mod \, 5$$

For $25 = 11001_2$

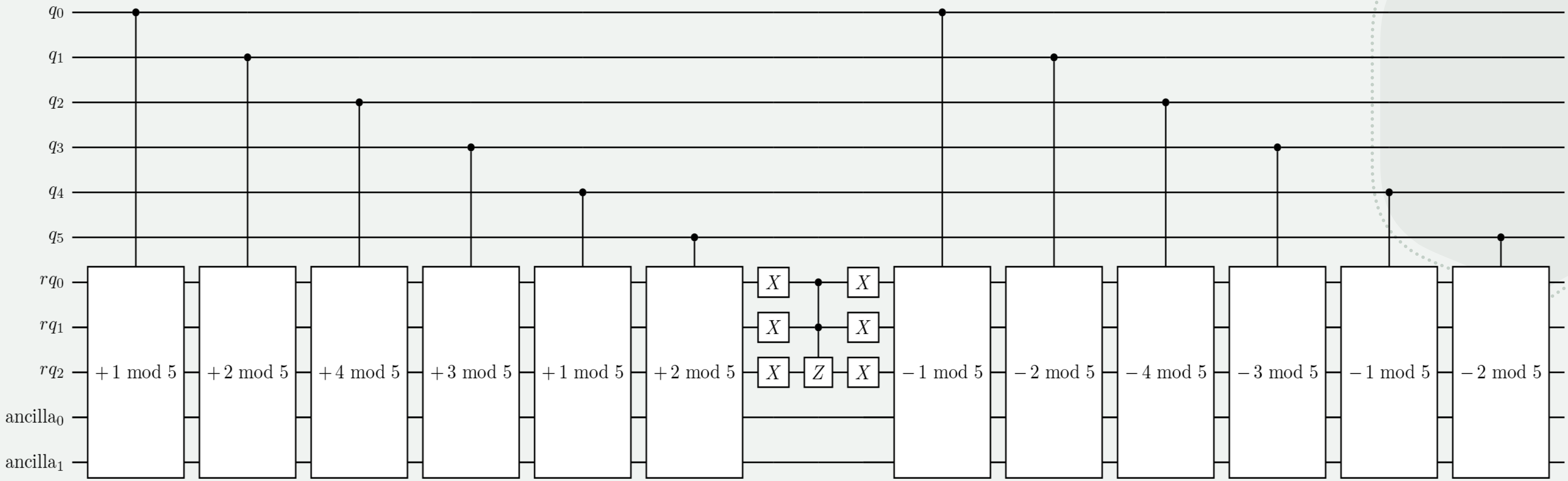
$$25 \equiv 0 \, mod \, 5$$

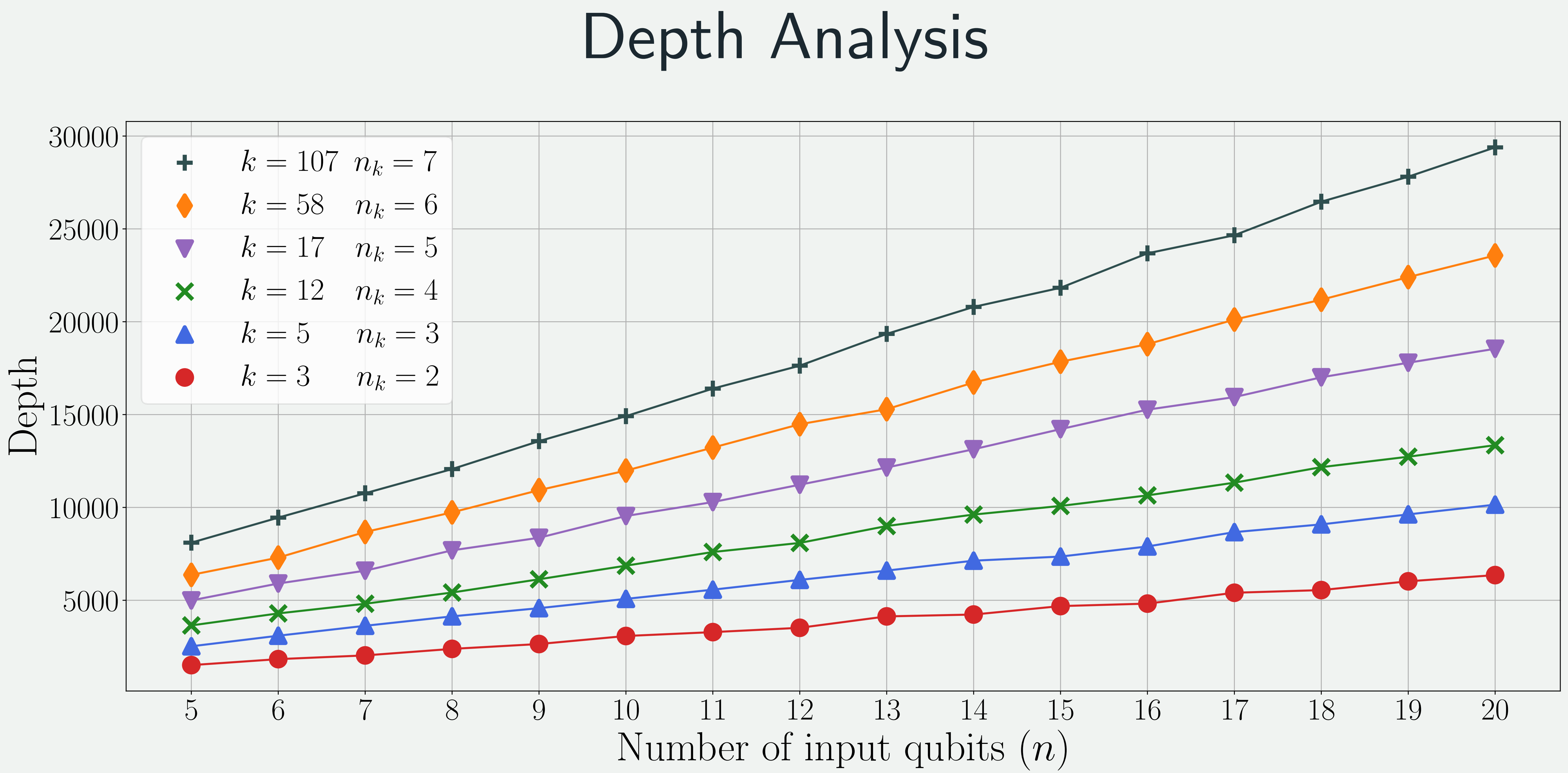
Building the quantum circuit



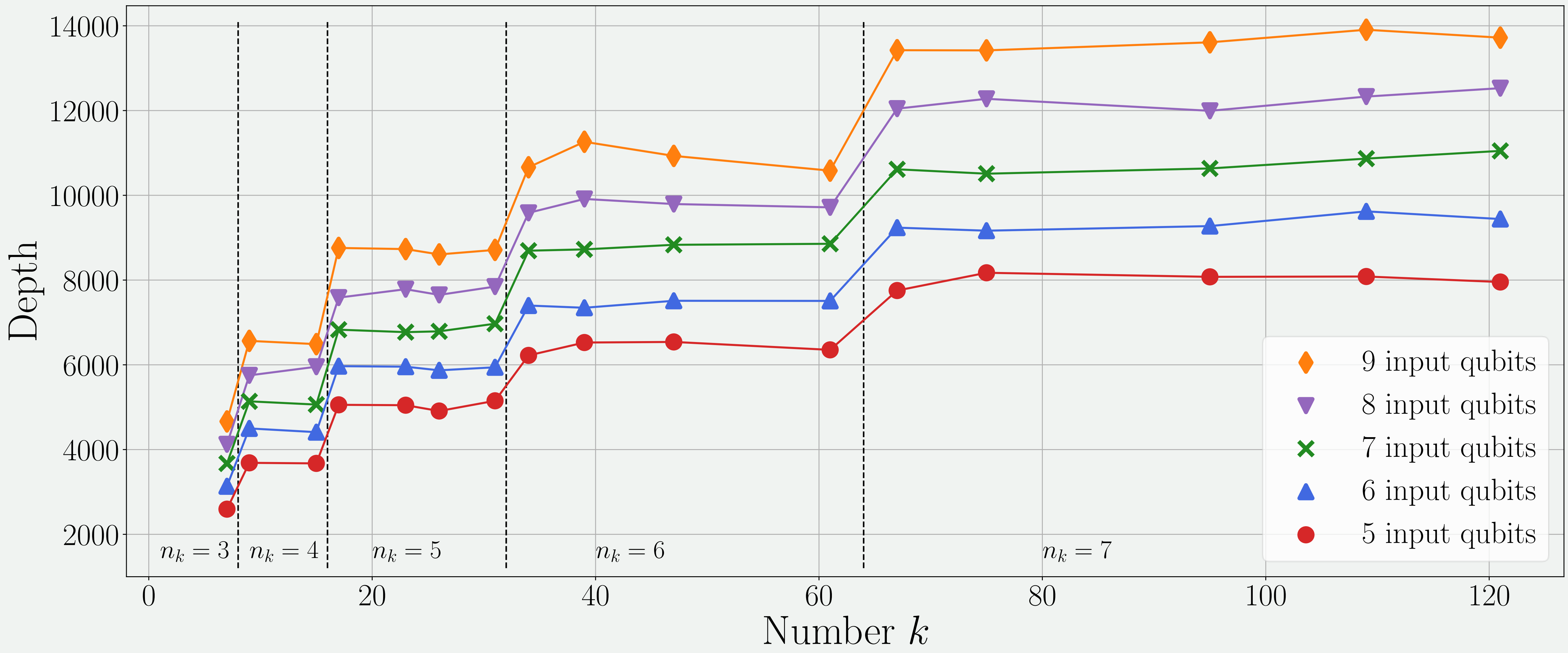
Example circuit: Multiples of $k = 5$

$$n = 6, \qquad N = 2^6 = 64, \qquad S = \{0, 1, \ldots, 62, 63\}$$




Depth Analysis
$k = 107$ $n_k = 7$
$k = 58$ $n_k = 6$
$k = 17$ $n_k = 5$
$k = 12$ $n_k = 4$
$k = 5$ $n_k = 3$
$k = 3$ $n_k = 2$
Depth
30000
25000
20000
15000
10000
5000
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
Number of input qubits (n)

Depth Analysis



Summary 🤝

- Presented ongoing research
- Multiples of oracle
- Algorithm for building the oracle given *k*
- Linear depth on the number of qubits $\mathcal{O}(n)$
- Classical computations $\mathcal{O}(n)$
- Code available: https://github.com/JSRivero/oracle-multiples

Future Work

- Creation of reusable quantum software for programmers
- Step in the creation of a bigger set of operations
- Explore other classical operations with integers
- Composable tools for creation of complex algorithms

Thank you for your attention

github.com/JSRivero

javier.sanchez@cenits.es