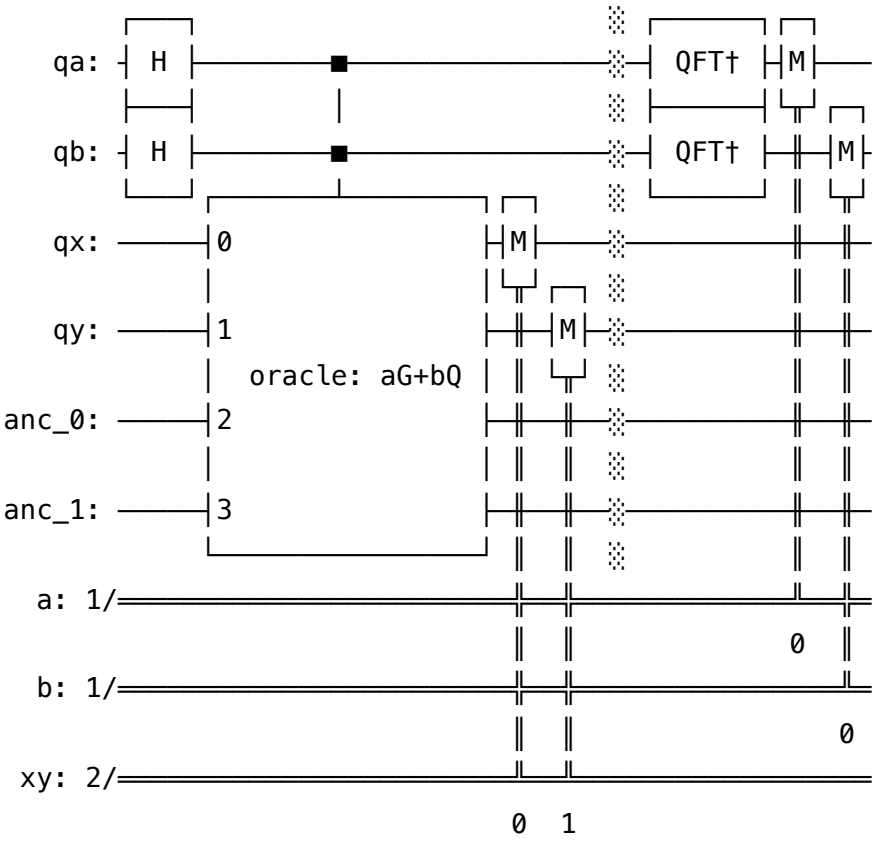
QDay Prize

Quantum Circuit Overview

We constructed the following quantum circuit based on the Proos–Zalka algorithm to break ECC cryptography.
Given the ECC parameter G and the public key $Q = dG$, we use Shor's algorithm with $aG + bQ$ as the Oracle.
For details of the Oracle, see this report.



The quantum circuit consists of the following steps:

1. Prepare qa and qb with the number of bits of the order n.
2. Apply H gates to qa and qb to create a superposition.
3. Execute the quantum circuit so that $(qx, qy) = aG + bQ$.
4. Measure (qx, qy) to collapse to a single coordinate.
5. Apply the inverse quantum Fourier transform ($QFT^{\dagger}$) to qa and qb.
6. Finally, measure qa and qb.

The private key is obtained from the measured values a and b as follows:

1. Calculate the number of bits len of the order n:

$$\mathrm{len} = \lfloor \log_2 n \rfloor$$

2. Scale a and b to fit the range 2^{len} to n:

$$x = a \cdot \frac{n}{2^{len}} \mapsto \begin{cases} \mathrm{floor}(a \cdot \frac{n}{2^{len}}) \\ \mathrm{ceil}(a \cdot \frac{n}{2^{len}}) \end{cases}$$
$$y = b \cdot \frac{n}{2^{len}} \mapsto \begin{cases} \mathrm{floor}(b \cdot \frac{n}{2^{len}}) \\ \mathrm{ceil}(b \cdot \frac{n}{2^{len}}) \end{cases}$$

   To account for rounding errors, all four combinations of floor/ceil for x and y are checked.
3. Compute the private key for all four combinations of floor/ceil for x and y:

$$d = x^{-1} \cdot y \mod n$$

   That is, calculate d for each of the four combinations (floor/floor, floor/ceil, ceil/floor, ceil/ceil) and check if any of them satisfy the private key condition.

From simulation, the probability of obtaining a measurement result that leads to the correct answer is about 60% to 70% when the number of bits of the order is small, but when the number of bits increases, the probability exceeds 80%.
On a real quantum computer (ibm_torino), due to noise, the result was almost random and the correct answer rate was similar to random guessing.

Required Resources

This work implements a general-purpose quantum circuit that works for any ECC parameter, so it requires a relatively large number of quantum gates.
For the ECC Curves and Keys presented, we implemented both a compact version (which reduces the number of qubits at the cost of deeper circuits) and a wide version (which uses more qubits but shallower circuits).
For details, see this report.

For example, to break secp256k1 used in Bitcoin, the compact version requires over 120,000 qubits, and the wide version requires nearly 30 million qubits.
Therefore, at present, it is considered unlikely that ECC can be broken by quantum computers.