Development of Novel Quantum Algorithms:

Classically verifiable quantum advantage from a computational Bell test

Filippo Tramonto (Italy)

Ban Tran (USA)

Nahid Binandeh Dehaghani (Portugal)

QUIP

Womanium Quantum+AI 2024

Classiq <> Womanium

AUGUST 2024

Problem Statement

Whether the correctness of the quantum computation is efficiently verifiable by a classical computer?

- **Sampling from entangled quantum many-body wavefunctions.**
- **Solving a deterministic problem via a quantum algorithm.**
- **Proving quantumness through interactive protocols.**

Project Solution

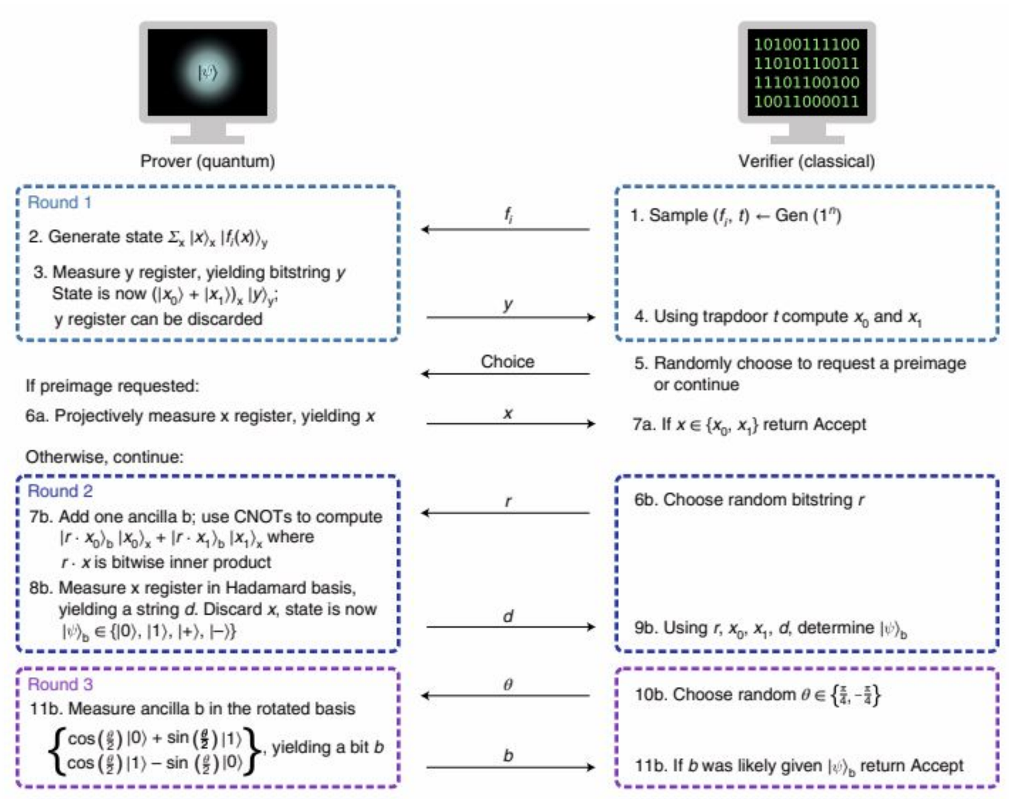


Image from: Kahanamoku-Meyer, Gregory D., et al. "Classically verifiable quantum advantage from a computational Bell test." *Nature Physics* 18.8 (2022).

Objective 1: Implement a toy problem of the algorithm: Phase Circuit "fast", in Qmod/Classiq Python SDK from the original paper code in Cirq.

Objective 2: Implement the whole protocol, including writing from scratch the quantum and classical functions

Objective 3: Estimate quantum resources and comparing them for different hardwares

Objective 4: Improve the implementation of the key quantum algorithm, for a better optimization

- **Quantum Algorithm:** Prepares the quantum state $|\psi\rangle = \sum_x |x\rangle_x |f_N(x)\rangle_y$ where $f_N(x) = x^2 \bmod N$ s the Rabin's TCF.
- **Quantum Circuits:** Utilizes phase circuits (working in Quantum Fourier space) for implementation.
- **Verification Protocol:** Implementing the paper's verification protocol, that is an interactive proof that relies on computational Bell tests, simplifying the cryptographic requirements while ensuring efficient verification by classical means.

〈 WOMANIUM | QUANTUM 〉

Project Step 1: implementation of the key quantum algorithm

We implemented the Gate-optimized Phase Circuit of the paper with Qmod/Classiq Python SDK

The circuit implement $f_N(x) = x^2 \bmod N$ in the state superposition $|\psi\rangle = \sum_x |x\rangle_x |f_N(x)\rangle_y$ using Quantum Phase Estimation

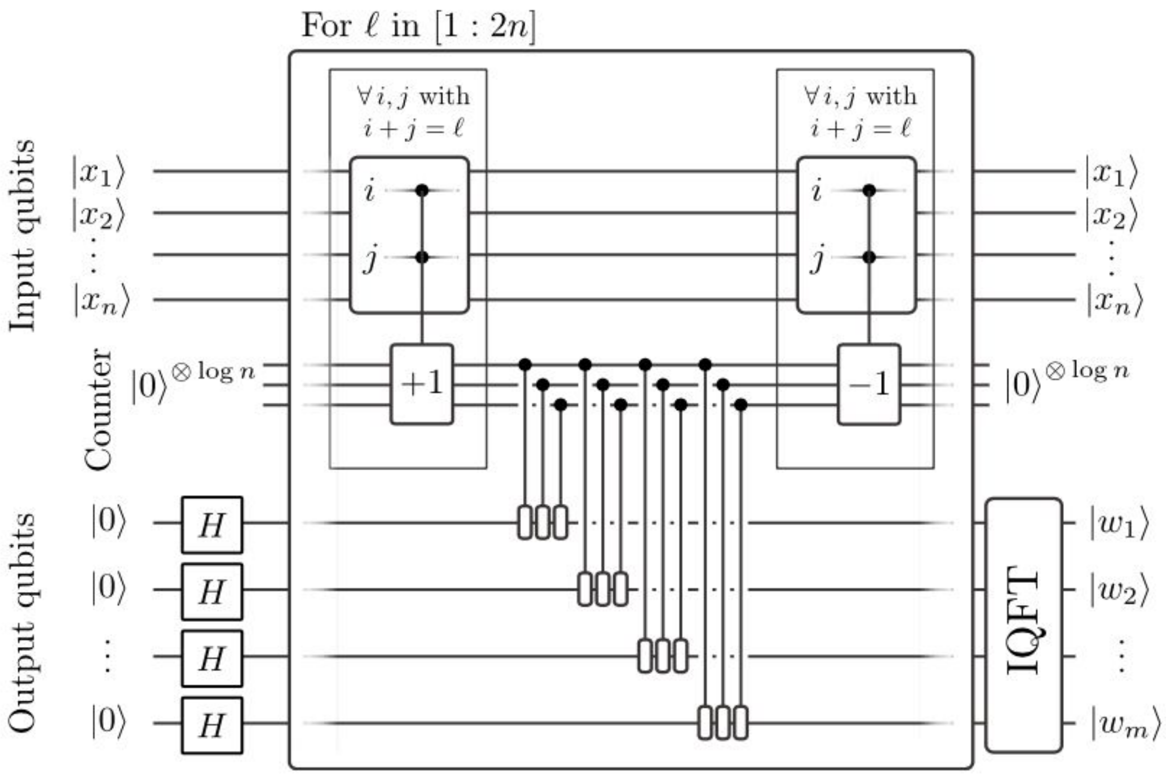


Image from: Kahanamoku-Meyer, Gregory D., et al. "Classically verifiable quantum advantage from a computational Bell test." Nature Physics 18.8 (2022).

Paper ➤ 4 quantum circuits: 2 digital, 2 phase circuits

We ➤ Gate optimized Phase Circuit

Key components of the algorithm:

1. **Controlled Phase Rotations** of pairs of qubits
2. The **Counter** is used to reduce the gate counts exploiting ancilla qubits and performing counts in the phases
3. **IQFT** transfers the the phases into output register

Implemented with the functions

- **x2_mod_N_phase() , x2modN_fast()**
- **count(), phase_add(), MCZPhase()**
- **qft(), iqft()**

Project Step 2: extension of the implementation

We implemented the whole protocol with Qmod/Classiq Python SDK

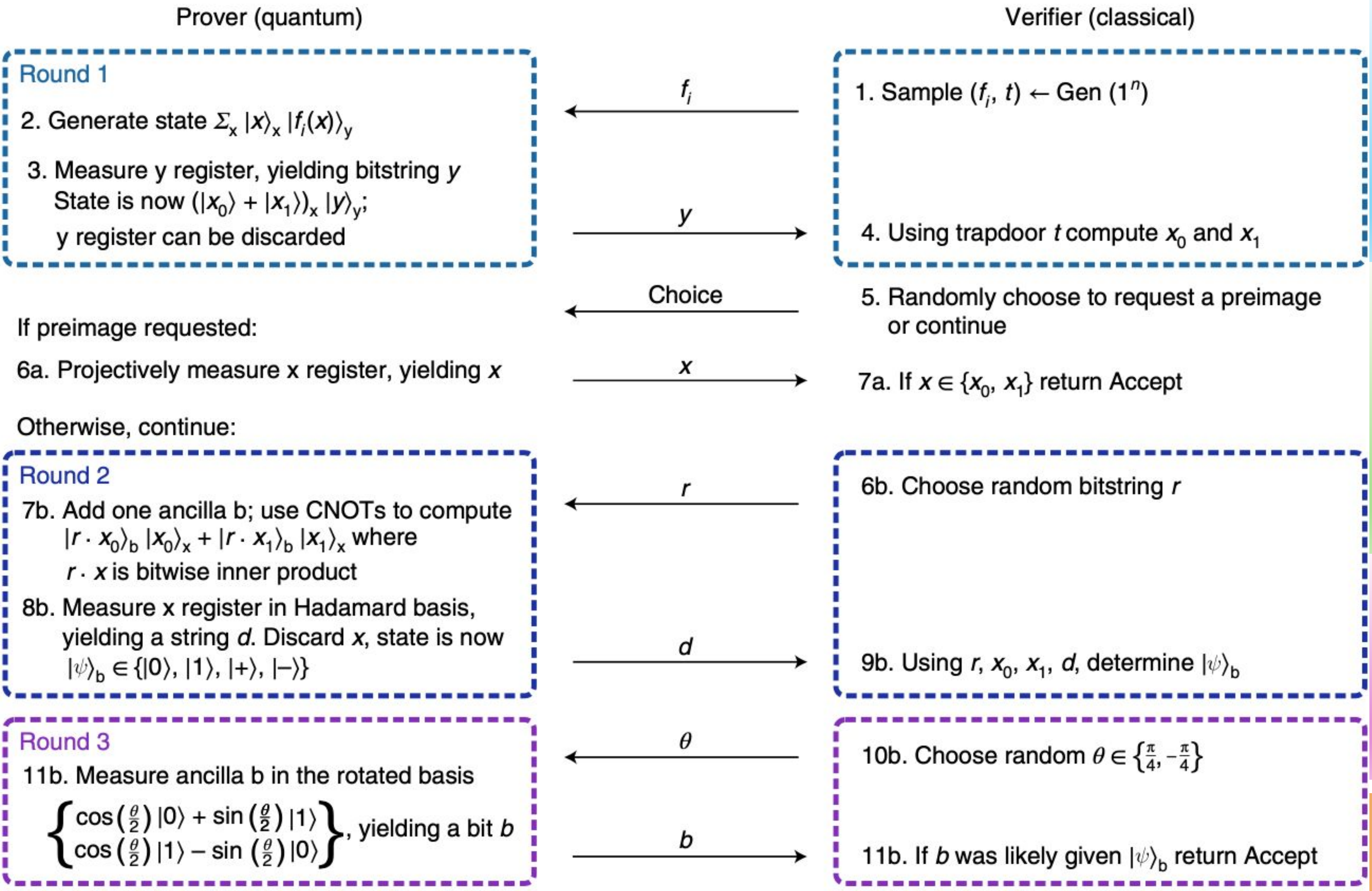


Image from: Kahanamoku-Meyer, Gregory D., et al. "Classically verifiable quantum advantage from a computational Bell test." *Nature Physics* 18.8 (2022).

Round 1:

- **gen_1lambda()**
- **x2_N_mod_phase()**
- **compute_x0x1()**

Round 2:

- **classical_bitwise_dot_prod_mod2()**
- **bitwise_inner_product_mod2()**
- **determine_state_psi_b()**

Round 3:

- **measure_ancilla_in_rotated_basis()**
- **final_verifier_validation()**

All measurements actually are performed at the end of the quantum side

Classiq's Generated Circuit Samples

2024-08-08T12:25:03.56...

Transpiled Info | Program Info | Data

Backend name: Default

Depth: 874

Width: 12

Gate count

U : 897

CX : 690

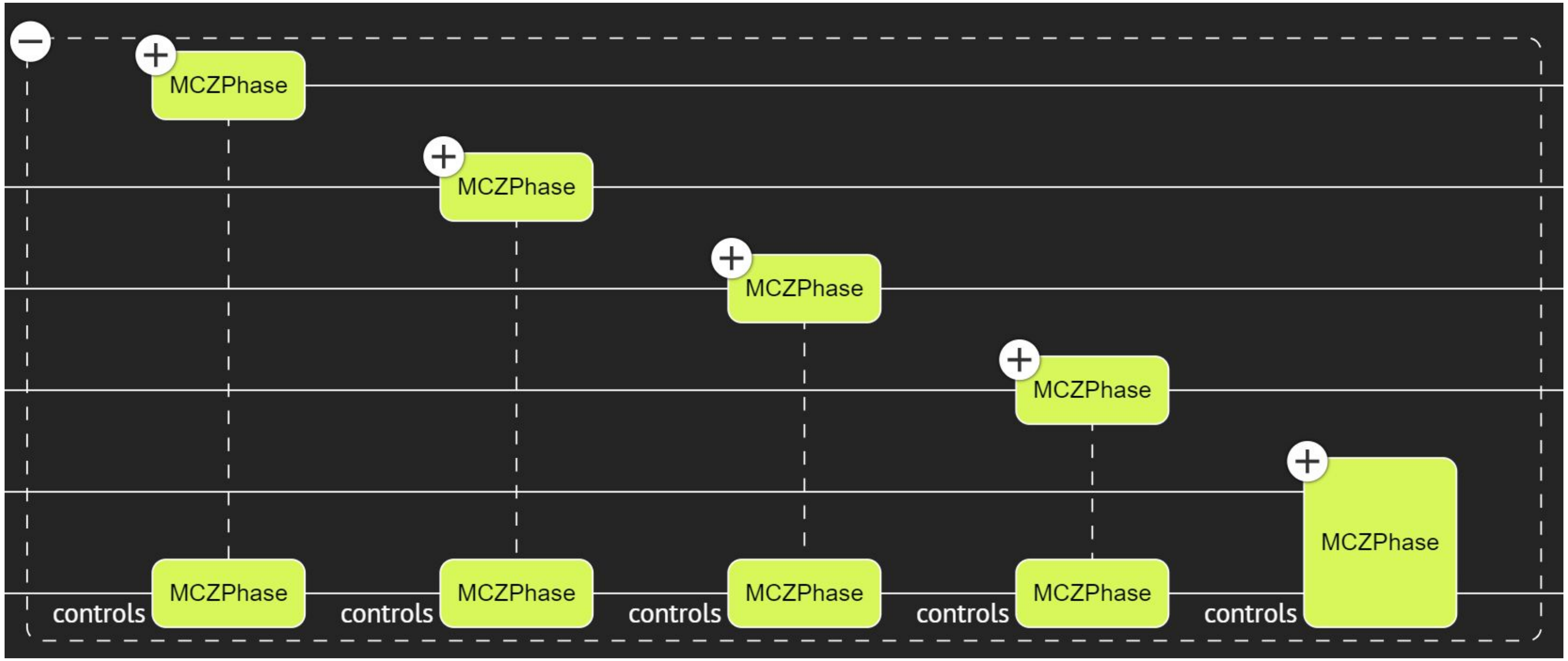
Hardware details

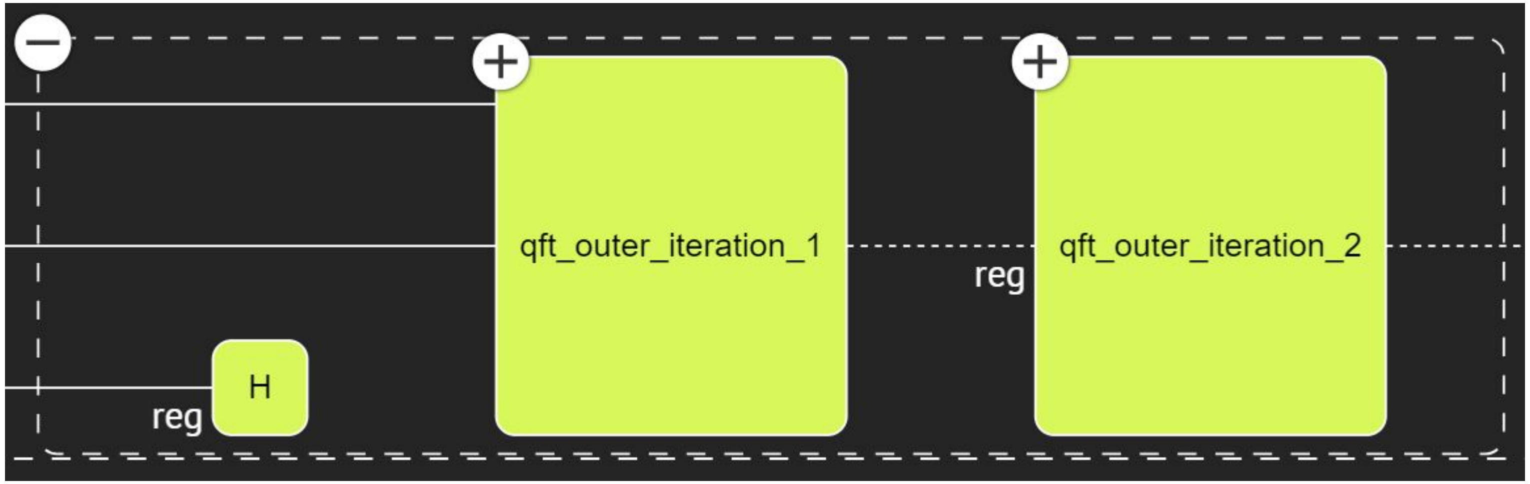
Basis Gates: cx, u

Connectivity Map:

null

Symmetric Connectivity: True





Challenges in Implementation

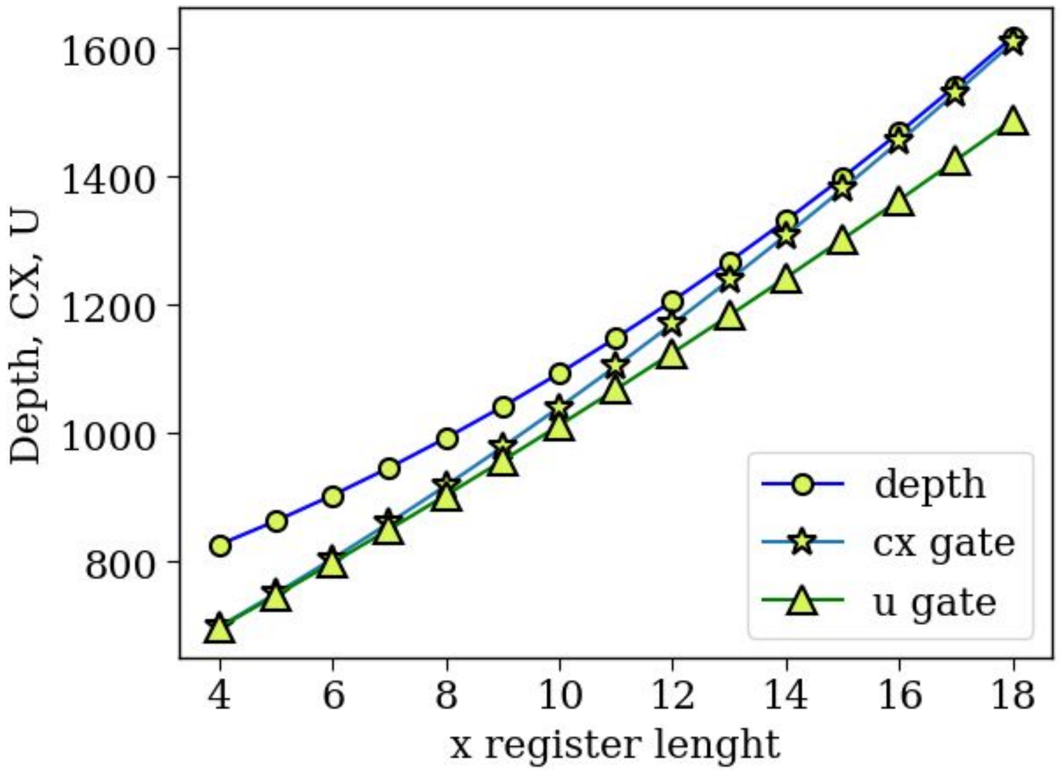
During the project development we met some challenges due to:

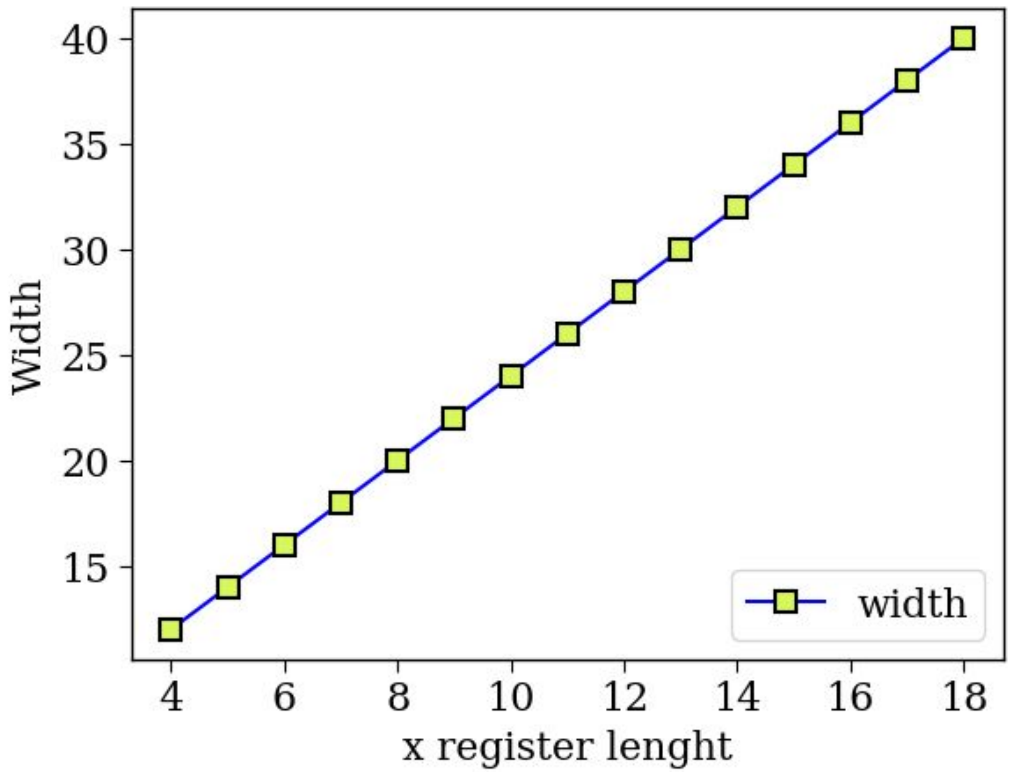
- Classiq’s Qmod language is a developing language→ Documentation and functions are changing and updating and in some case the documentation is missing
- Different characteristics between Cirq and Qmod such as the endianness.
- The APIs are changing → Versions are increased in a short time. For example, we had to update our previous source code.
- Integration with other current Quantum Tools such QREs are still in the air.

Project Step 2: Resource Estimation

Quantum resources were measured by synthesizing the quantum program of the phase circuit. The data presented here pertains to the transpiled information, independent of specific hardware constraints.

$2n^2 \log n + \mathcal{O}(n^2)$ gates.





Future Scope

- Improve our implementation in Qmod/Classiq Python SDK
- Implement the three alternative quantum circuits proposed in the paper using Qmod/Classiq Python SDK.
- Implement variants of the Phase circuits, that differ in terms ancillas utilizations and gate counts.
- Perform further Quantum Resource Estimation of the quantum program developed with Qmod/Classiq Python SDK.

Thank YOU.