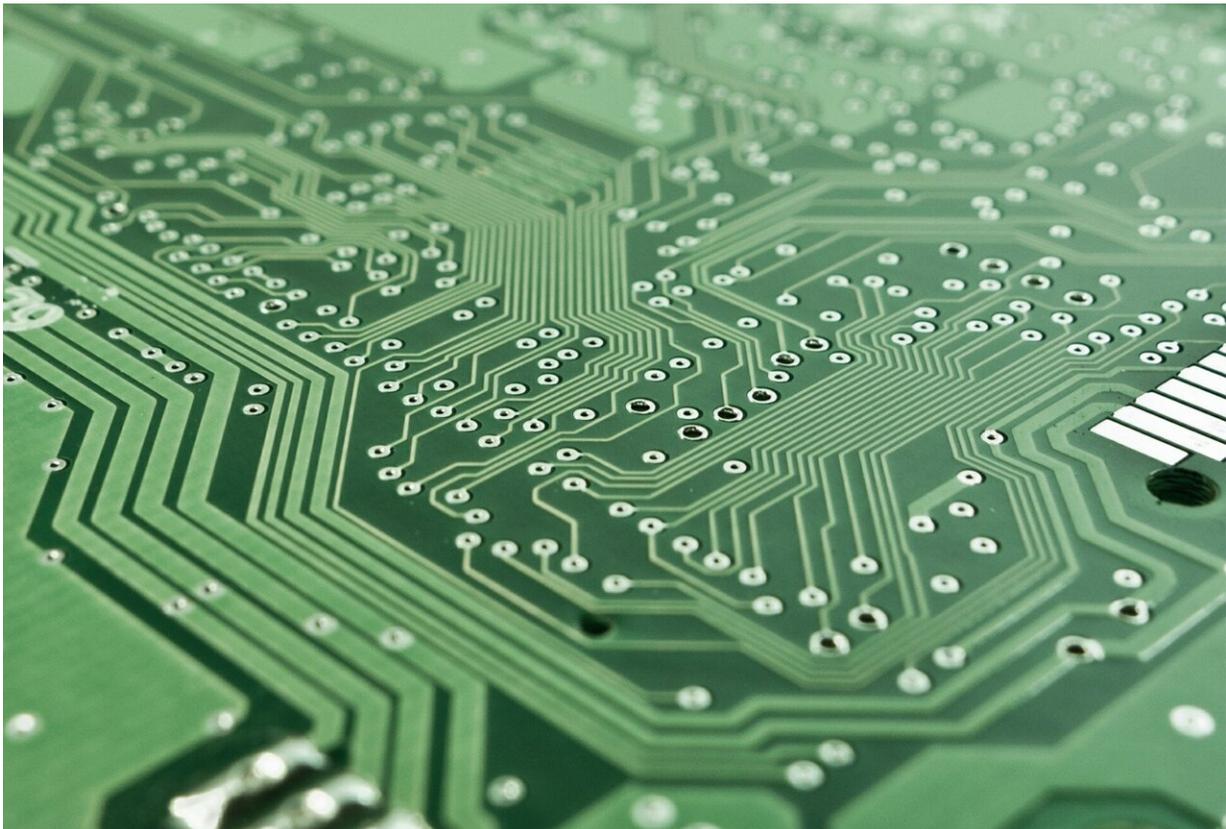


Researchers explore architectural design of quantum computers

June 26 2019, by Prakash Murali, Margaret Martonosi



Credit: CC0 Public Domain

A recent study led by Princeton University researchers, in collaboration with University of Maryland and IBM, explored the architectural design of quantum computers (QC). In a [paper](#) presented at the 2019

ACM/IEEE International Symposium on Computer Architecture, the researchers performed the largest real-system evaluation of quantum computers to date, using seven quantum computers from IBM, Rigetti and the University of Maryland.

The researchers developed new software to compile from QC applications to [hardware](#) prototypes; on error-prone early-stage QC hardware, this compiler delivers up to 28 times improvement in program correctness rates compared to industry compilers. The study stresses the importance of careful instruction set design, rich connectivity topologies and the need to co-design applications and hardware to achieve the best performance from fledgling QC systems.

Widely different quantum computing technologies

Quantum computing is a fundamentally new paradigm of computation with promising applications in drug design, fertilizer design, artificial intelligence and secure information processing, among other things. From its inception in the 1980s as a purely theoretical endeavor, [quantum computing](#) has now progressed to the point that small prototype systems are available for experiments. Companies such as IBM and Rigetti now offer free access to their five- to 16-qubit systems over the cloud. These systems can be programmed using sequences of instructions, also known as operations or gates.

Analogous to the early days of classical computing involving systems built with vacuum tubes relay circuits or transistors, QC systems today can be built out of several hardware technologies. Front-runner technologies include [superconducting qubits](#) and the trapped ion qubits, with other candidate technologies also of considerable interest. However, unlike classical binary computers, QC technologies are so different that even the fundamental gate operations that can be performed on a single qubit differ widely. Selecting the most appropriate gate operations to be

exposed for software use is an important QC design decision.

Current QC systems also differ in terms of the reliability of operations between pairs of qubits. For example, in superconducting qubits such as those from IBM and Rigetti, the qubits are printed on a 2-D wafer using a method similar to classical processor fabrication. In these systems, inter-qubit operations are permitted only between qubits that are close to each other and connected by special wires. This fabrication approach imposes limitations in how different qubits can communicate, namely allowing each qubit in the system to interact directly with only a few other near-neighbor qubits. In contrast, for the trapped ion qubits in UMD, inter-qubit operations are accomplished using the vibrational motion of a chain of ions. Because this approach does not use physical connections in the form of wires, it allows inter-qubit operations between any pair of qubits in the system. This more expansive communication model can be helpful to some QC algorithms.

A third characteristic of note is that in all these candidate technologies, the [quantum state](#) is very hard to manipulate precisely. This leads to operational error rates. In addition, the magnitude of these errors varies significantly, both across the qubits in the system and across time. As a result, these large noise variations change the reliability of the operations by up to a factor of 10. Since QC algorithms chain several of these operations together, per-operation error rates compound to make it hard for the program to get the right answer overall.

Architecture for quantum computers

The dramatic differences among different QC implementations has spurred researchers to design programming interfaces that shield the programmer from the implementation details and error rates of the qubits. Such an interface, commonly known as instruction set architecture (ISA), serves as a cornerstone of modern computing

systems.

The ISA includes a set of instructions that can be executed on the hardware and serves as a contract between the hardware implementation and software. As long as the program uses operations permitted by the ISA, it can be run without modifications on any hardware, which also implements the same ISA, irrespective of any differences among the hardware implementations.

QC vendors make a number of design decisions about the ISA and the connectivity of the qubits. Each vendor chooses to provide a set of software-visible gates that mask the particular details of the gate implementations. These gates are typically not the same as the fundamental operations, and are commonly chosen to be operations that are frequently used by quantum algorithm designers and programmers.

"Which gates should a vendor choose to expose to hardware? Should we abstract these gates in a common ISA across vendors or tailor them to the underlying device characteristics?" asks Prakash Murali, a graduate student at Princeton and an author on the study. Similarly, the choice of qubit connectivity, although influenced by the hardware technology, determines what inter-qubit operations a program can use. "How should the vendor interconnect their qubits? How does the interplay of variable noise rates and connectivity influence programs?" says Murali.

Design insights for quantum computer architecture

To answer these design questions, the researchers evaluated the [architecture](#) of seven systems from three vendors, IBM, Rigetti and UMD, with different connectivity topologies, spanning two hardware [qubit](#) technologies. Since quantum computers are noisy, it is standard practice in the field to run programs several thousand times and report the answer which occurs most frequently as correct answer. To increase

the likelihood of correct runs, this work developed TriQ, a multi-vendor optimizing compiler that outperforms vendor compilers by significant margins, despite cross-platform applicability.

Using TriQ, the researchers showed that the architectural design choices of a system can significantly influence the correctness rate of program runs, underscoring the importance of making these design choices with program requirements in mind. They observed that the vendor's choice for the software-visible gate set can influence both the number of operations required to run a program and the correctness rate. When the vendor exposes the native or fundamental operations, TriQ can significantly reduce the number of native operations required to perform a set of program instructions, and increase the correctness rate. This suggests that on quantum computers, it is premature to shield all knowledge of the native instructions through a device or vendor-independent ISA in a manner similar to classical systems.

"We also found that the match between the application's communication requirements and the hardware connectivity topology is crucial. When the hardware can support an application with only a small number of communication operations, the application usually has higher chances of executing correctly. When there is a mismatch, and a lot of communication operations are required, application correctness rates suffer," said Murali.

With its open-source software tools now available on [github](#), the work in this paper has the potential to offer significant real-world improvements in QC software compilation, while also offering the opportunity for broader insights on the design approaches most effective for QC hardware.

More information: Full-Stack, Real-System Quantum Computer Studies: Architectural Comparisons and Design Insights,

arXiv:1905.11349 [quant-ph] arxiv.org/abs/1905.11349

Provided by Princeton University

Citation: Researchers explore architectural design of quantum computers (2019, June 26)
retrieved 18 September 2024 from

<https://phys.org/news/2019-06-explore-architectural-quantum.html>

This document is subject to copyright. Apart from any fair dealing for the purpose of private study or research, no part may be reproduced without the written permission. The content is provided for information purposes only.