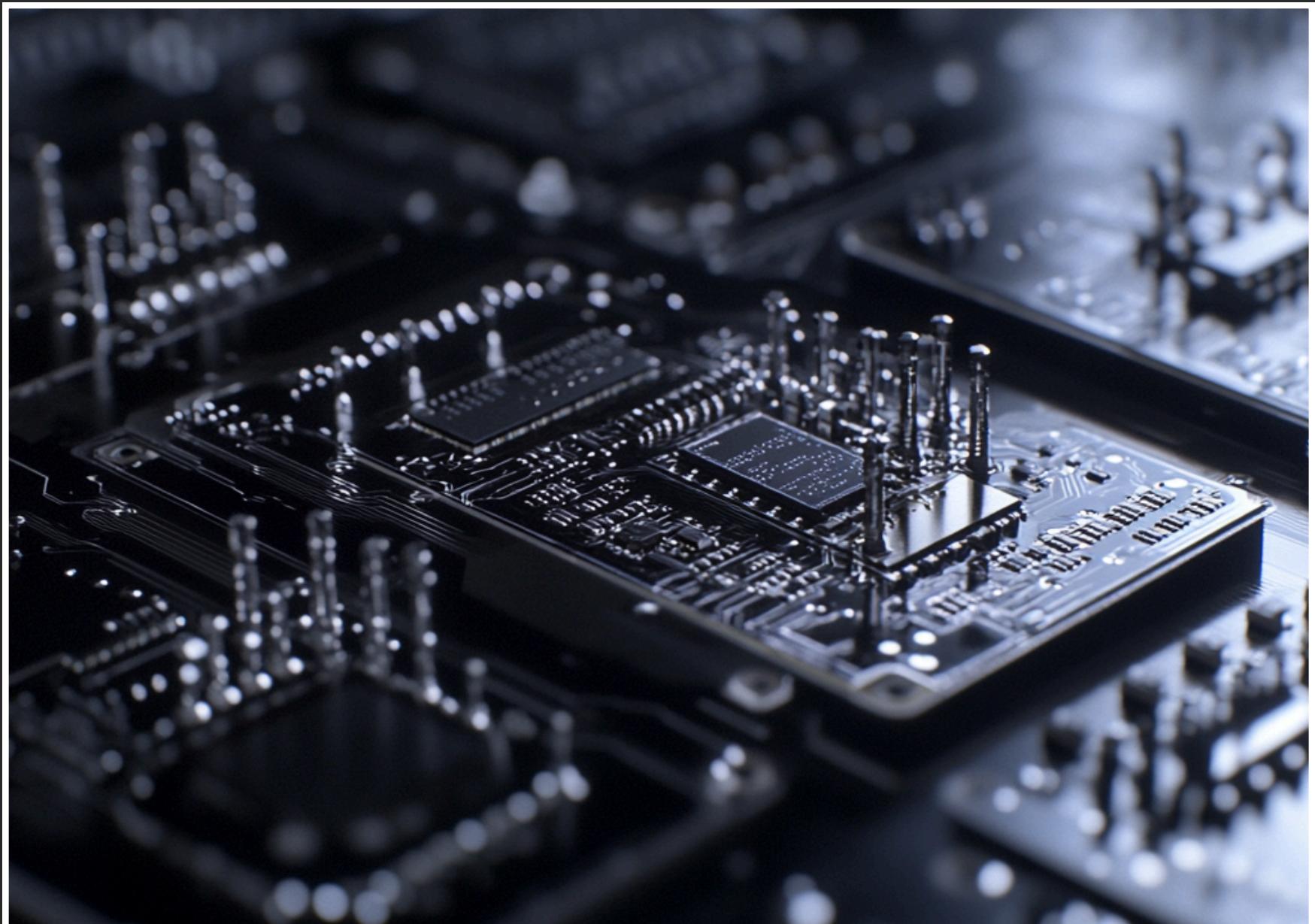


# 99% gate fidelity

Is that all we need? Or is there more to the story?



You've probably heard that we need **99% fidelity** for fault tolerant quantum computing.

But where does that number come from?

What does it mean?

And is it the whole story? 🤔

# Where does the 99% number come from?

99% is a widely-accepted rule of thumb 

It's based on insights from research & simulations that simplify complex error thresholds.



**Article**

**Quantum logic with spin qubits crossing the surface code threshold**

---

https://doi.org/10.1038/s41586-021-04273-w

Received: 1 July 2021

Accepted: 22 November 2021

Published online: 19 January 2022

Open access

---

Xiao Xue<sup>1,2</sup>, Maximilian Russ<sup>1,2</sup>, Nodar Samkhnadze<sup>1,2</sup>, Brennan Endseth<sup>1,2</sup>, Amir Sammak<sup>1,2</sup>, Giordano Scappucci<sup>1,2</sup> & Lieven M. K. Vandersypen<sup>1,2\*</sup>

High-fidelity control of quantum bits is paramount for the reliable execution of quantum algorithms and for achieving fault tolerance—the ability to correct errors faster than they occur<sup>1</sup>. The central requirement for fault tolerance is expressed in terms of an error threshold. Whereas the actual threshold depends on many details, a common target is the approximately 1% error threshold of the well-known surface code<sup>2,3</sup>. Reaching two-qubit gate fidelities above 99% has been a long-standing major goal for semiconductor spin qubits. These qubits are promising for scaling, as they can leverage advanced semiconductor technology<sup>4</sup>. Here we report a spin-based quantum processor in silicon with single-qubit and two-qubit gate fidelities, all of which are above 99.5%, extracted from gate-set tomography. The average single-qubit gate fidelities remain above 99% when including crosstalk and diling errors on the neighbouring qubit. Using this high-fidelity gate set, we execute the demanding task of calculating molecular ground-state energies using a variational quantum

**PHYSICAL REVIEW X 10, 011022 (2020)**

---

**Topological and Subsystem Codes on Low-Degree Graphs with Flag Qubits**

Christopher Chamberland<sup>1,2\*</sup>, Guanyu Zhu<sup>3,4</sup>, Theodore J. Yoder<sup>1</sup>, Jared B. Hertzberg, and Andrew W. Cross<sup>1</sup>  
IBM T. J. Watson Research Center, Yorktown Heights, New York 10598, USA

(Received 30 July 2019; revised manuscript received 28 October 2019; published 31 January 2020)

In this work we introduce two code families, which we call the heavy-hexagonal code and the heavy-square code. Both code families are implemented by assigning physical data and ancilla qubits to both vertices and edges of low-degree graphs. Such a layout is particularly suitable for superconducting qubit architectures to minimize frequency collisions and cross talk. In some cases, frequency collisions can be reduced by several orders of magnitude. The heavy-hexagonal code is a hybrid surface and Bacon-Shor code mapped onto a (heavy-) hexagonal lattice, whereas the heavy-square code is the surface code mapped onto a (heavy-) square lattice. In both cases, the lattice includes all the ancilla qubits required for fault-tolerant error correction. Naively, the limited qubit connectivity might be thought to limit the error-correcting capability of the code to less than its full distance. Therefore, essential to our construction is the use of flag qubits. We modify minimum-weight perfect-matching decoding to efficiently and scalably incorporate information from measurements of the flag qubits and correct up to the full code distance while respecting the limited connectivity. Simulations show that high threshold values for both codes can be obtained using our decoding protocol. Further, our decoding scheme can be adapted to other topological code families.

DOI: 10.1103/PhysRevX.10.011022

Subject Areas: Condensed Matter Physics, Quantum Physics, Quantum Information

**LETTER**

doi:10.1038/nature13171

**Superconducting quantum circuits at the surface code threshold for fault tolerance**

---

R. Barends<sup>1,\*</sup>, J. Kelly<sup>1,\*</sup>, A. Megrant<sup>1</sup>, A. Veitia<sup>2</sup>, D. Sank<sup>1</sup>, E. Jeffrey<sup>1</sup>, T. C. White<sup>1</sup>, J. Mutus<sup>1</sup>, A. G. Fowler<sup>1,3</sup>, R. Campbell<sup>2</sup>, Y. Chen<sup>1</sup>, Z. Chen<sup>1</sup>, B. Chiaro<sup>1</sup>, A. Dunsworth<sup>1</sup>, C. Neill<sup>1</sup>, P. O’Malley<sup>1</sup>, P. Roushan<sup>1</sup>, A. Vainsencher<sup>1</sup>, J. Wenner<sup>1</sup>, A. N. Korotkov<sup>4</sup>, A. N. Cleland<sup>1</sup> & John M. Martinis<sup>1</sup>

A quantum computer can solve hard problems, such as prime factoring<sup>5,6</sup>, database searching<sup>7,8</sup> and quantum simulation<sup>9</sup>, at the cost of needing to protect fragile quantum states from error. Quantum error correction<sup>10</sup> provides this protection by distributing a logical state among many physical quantum bits (qubits) by means of quantum entanglement. Superconductivity is a useful phenomena in this regard, because it allows the construction of large quantum circuits and is compatible with microfabrication. For superconducting qubits, the surface code approach to quantum computing<sup>11</sup> is a natural choice for error correction, because it uses only nearest-neighbour coupling and rapidly cycled entangling gates. The gate fidelity requirements are modest: the per-step fidelity threshold is only about 99 per cent. Here we demonstrate a universal set of logic gates in a superconduc-

maintaining a high level of coherence (see Supplementary Information for decoherence times). Here the four legs of the cross allow for a natural segmentation of the design into coupling, control and readout. We choose a modest inter-qubit capacitive coupling strength of  $g_{12} = 30$  MHz and use alternating qubit idle frequencies of 5.5 and 4.7 GHz, enabling a controlled-phase gate in 40 ms when two qubits are brought near resonance, while minimizing the effective coupling to 0.3 MHz when the qubits are at their idle points. Rotations around the X and Y axes in the Bloch sphere representation are performed using pulses on the microwave (XY) line, whereas Z-axis rotations, which control the phase of the quantum state, are achieved by a flux-bias current on the frequency-control (Z) line. We use a dispersive measurement method<sup>12</sup> whereby each qubit is coupled to a readout resonator with a distinct resonance

**PHYSICAL REVIEW A 86, 032324 (2012)**

---

**Surface codes: Towards practical large-scale quantum computation**

Austin G. Fowler  
Centre for Quantum Computation and Communication Technology, School of Physics, The University of Melbourne, Victoria 3010, Australia

Matteo Mariantoni, John M. Martinis, and Andrew N. Cleland  
Department of Physics, University of California, Santa Barbara, California 93106-9530, USA  
and California Nanosystems Institute, University of California, Santa Barbara, California 93106-9530, USA  
(Received 2 August 2012; published 18 September 2012)

This article provides an introduction to surface code quantum computing. We first estimate the size and speed of a surface code quantum computer. We then introduce the concept of the stabilizer, using two qubits, and extend this concept to stabilizers acting on a two-dimensional array of physical qubits, on which we implement the surface code. We next describe how logical qubits are formed in the surface code array and give numerical estimates of their fault tolerance. We outline how logical qubits are physically moved on the array, how qubit braid transformations are constructed, and how a braid between two logical qubits is equivalent to a controlled-NOT. We then describe the single-qubit Hadamard,  $\hat{S}$  and  $\hat{T}$  operators, completing the set of required gates for a universal quantum computer. We conclude by briefly discussing physical implementations of the surface code. We include a number of Appendices in which we provide supplementary information to the main text.

DOI: 10.1103/PhysRevA.86.032324

PACS number(s): 03.67.Lx

**I. BACKGROUND**

Quantum computers provide a means to solve certain problems that cannot be solved in a reasonable period of time using a conventional classical computer. These problems

transformations on a single surface, a highly significant simplification [12–14]. These authors also evaluated error tolerances for a fully planar implementation using only one- and two-qubit nearest-neighbor gates, arriving at an error

# What does it mean?

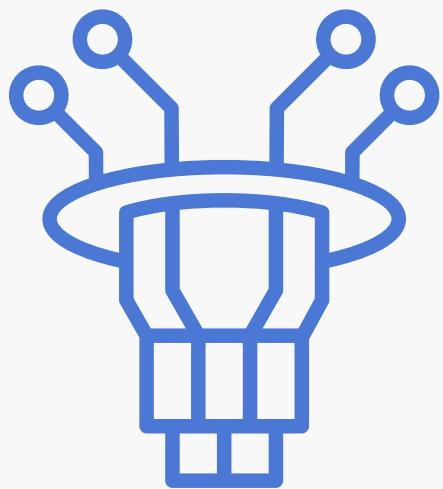
This rule of thumb says that once two-qubit gates reach 99% fidelity:

**Fault-tolerance becomes feasible!**

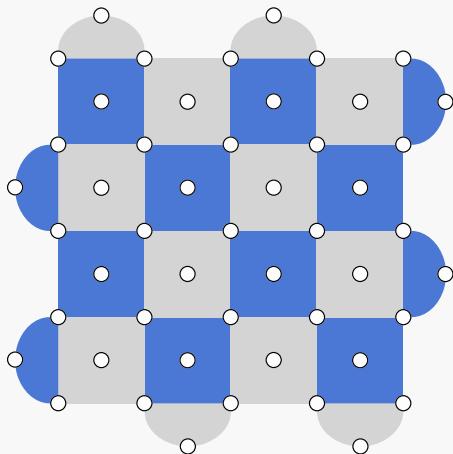


But that's only part of the quantum computing story.

The research and simulations we just mentioned only considered **superconducting qubits**.



And they used a specific error correction code called the **surface code**.



To advance quantum computing, we must explore new architectures.

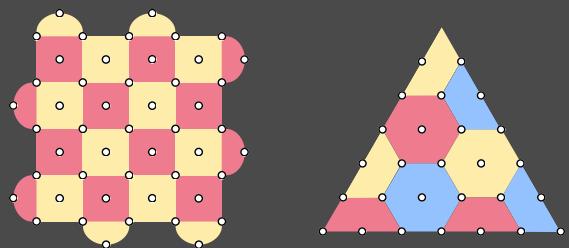
But how can we know their thresholds?

We start by defining the fault-tolerant architecture that we want to explore.

An architecture includes these four components:

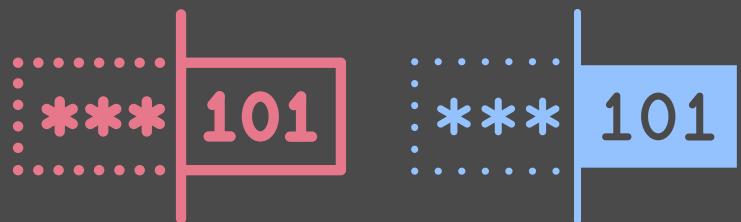
### Error Correction Code

How to connect qubits together.



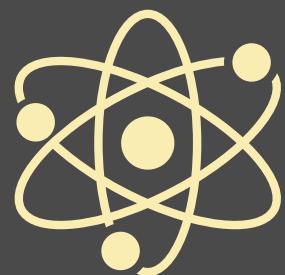
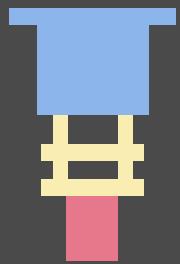
### Error Decoding

How to identify and fix errors.



### Qubit Definition

Which hardware to define the qubits.



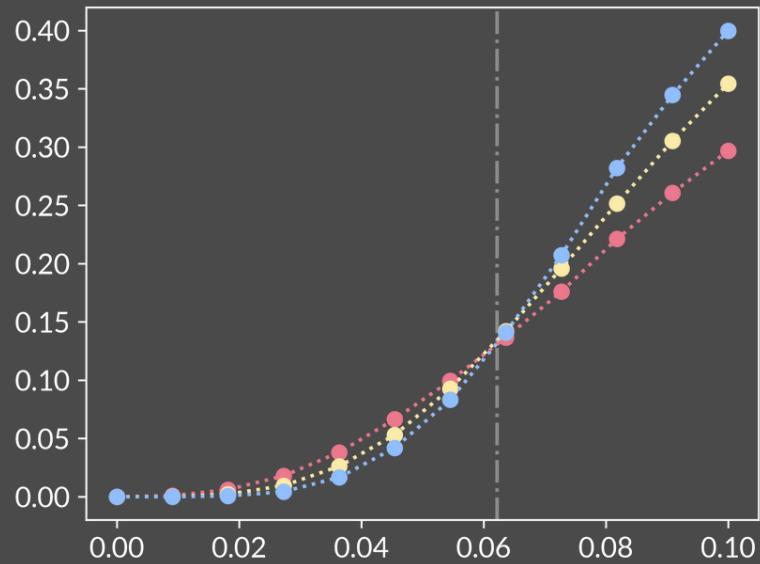
### Qubit Control

How to control the qubits.

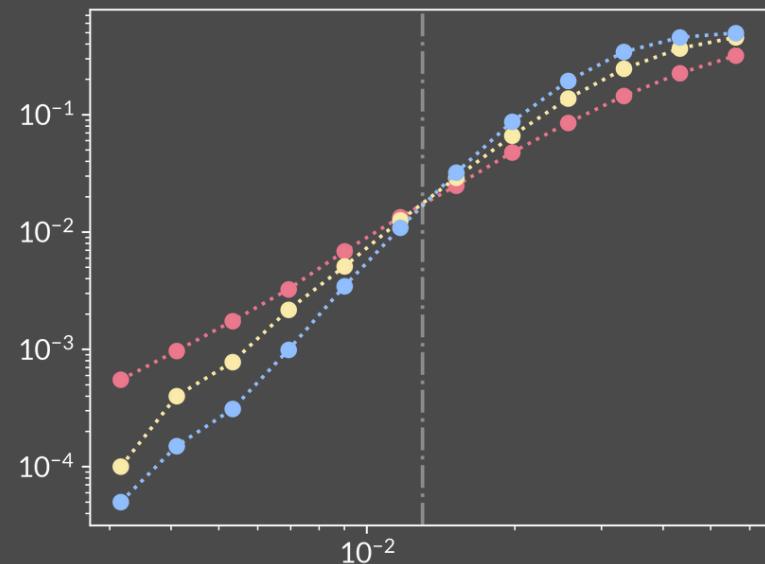


Then we compute threshold plots for different kinds of error models.

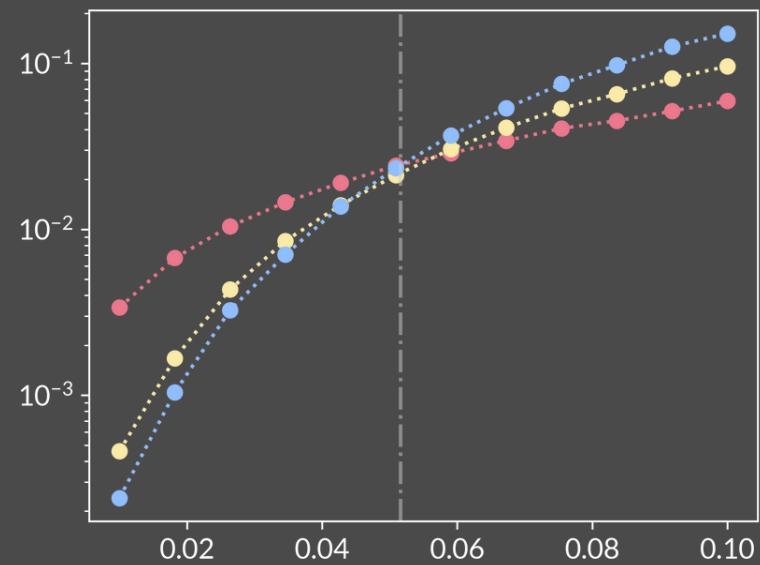
Depolarizing noise



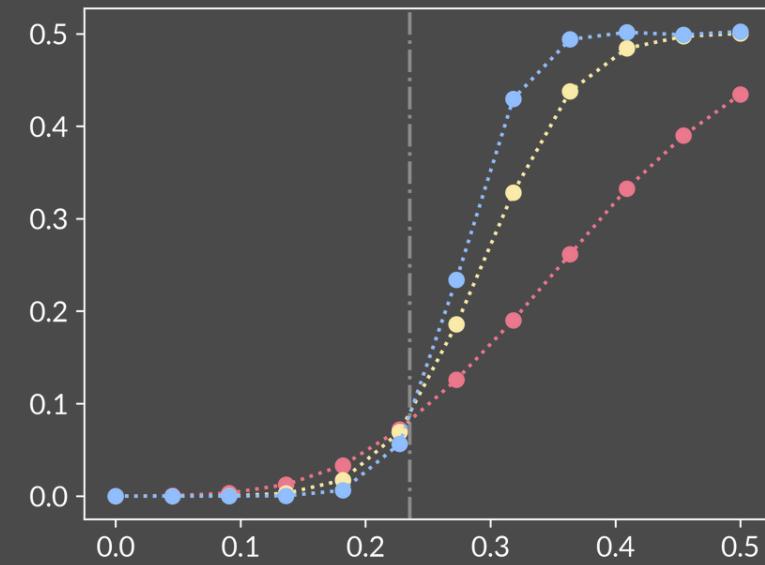
Mitigated leakage



Scattering



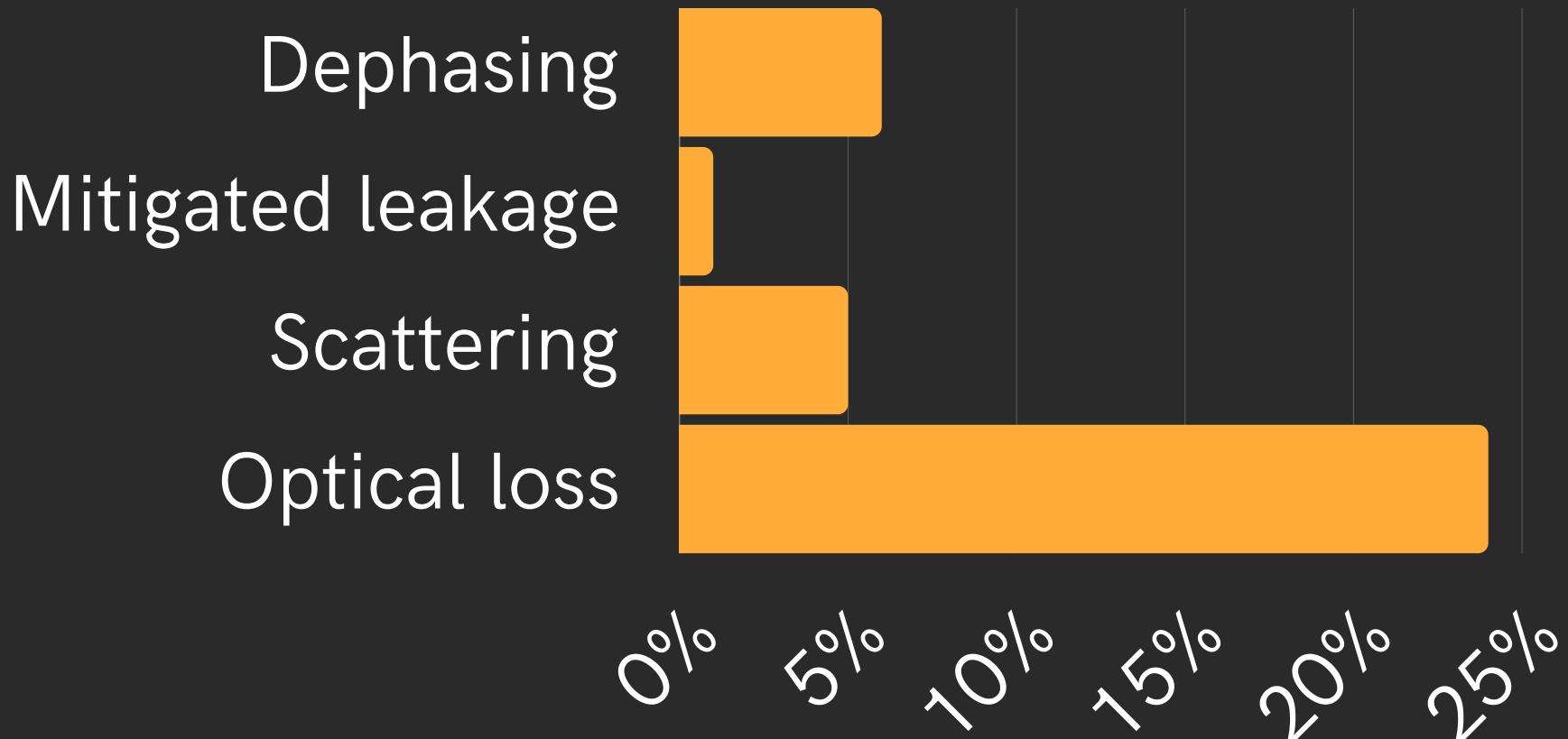
Optical loss



...and others as well!

# Fault tolerance thresholds

From these plots, hardware teams can extract the fault tolerance thresholds for different errors



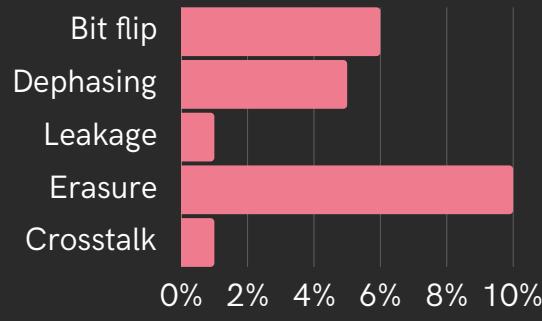
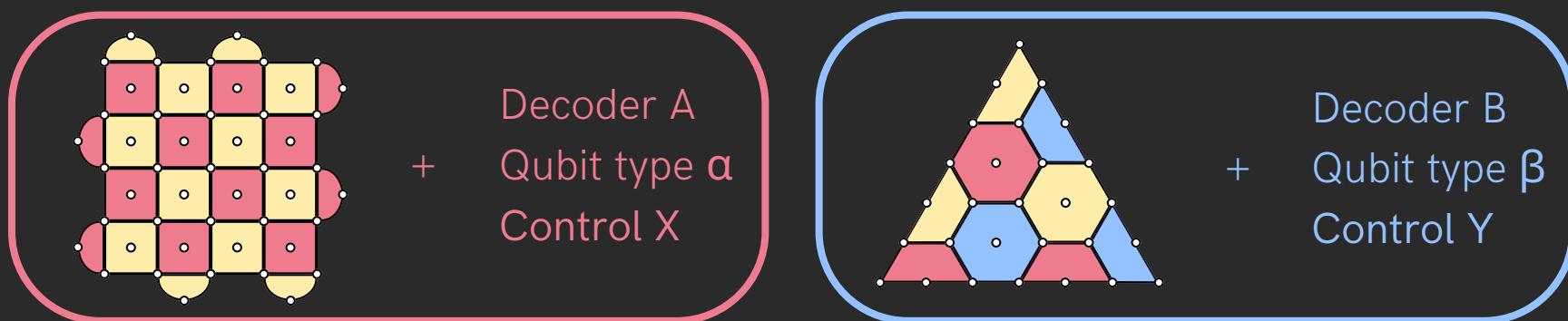
The key is to design and build architectures below the fault-tolerance thresholds **for all error types**.

But calculating fault-tolerance thresholds across various hardware imperfections and various architectures is **really, really hard!**

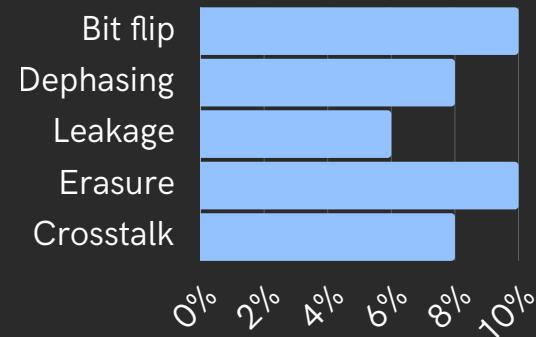
# That's why we built Plaquette™!

Plaquette simplifies threshold calculations.

Making it easier for quantum hardware teams to map out their hardware roadmap.



6 years away



4 years away!



# Learn more about Plaquette™

If you're interested to learn how to use Plaquette to plan out your hardware roadmap, we're happy to chat.

Contact us here on LinkedIn™, or head on over to the website at qc.design.

