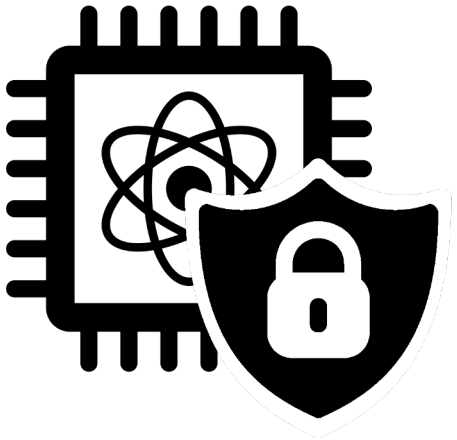


Tutorial on Security of Quantum Computing Systems

Quantum Week Tutorial

August 2025

Prof. Jakub Szefer (jakub.szefer@northwestern.edu)

Electrical and Computer Engineering

Northwestern University

Computer Architecture and Security Lab (CASLAB)

Northwestern University

Tutorial on Security of Quantum Computing Systems

This tutorial will introduce the audience to the field of quantum computer cybersecurity, which focuses on research on how to make quantum computing systems secure

- **Goals and contents of the tutorial:**

1. First, this tutorial will **introduce audience to classical computer security ideas and research topics** such as confidentiality, integrity, and availability, information leaks, side- and covert-channel attacks, or fault-injection attacks

2. Second, this tutorial will apply classical computer security mindset to quantum computers, and **demonstrate examples of security attacks** prototyped on real cloud-based **Noisy Intermediate-Scale Quantum (NISQ) quantum computers** available today

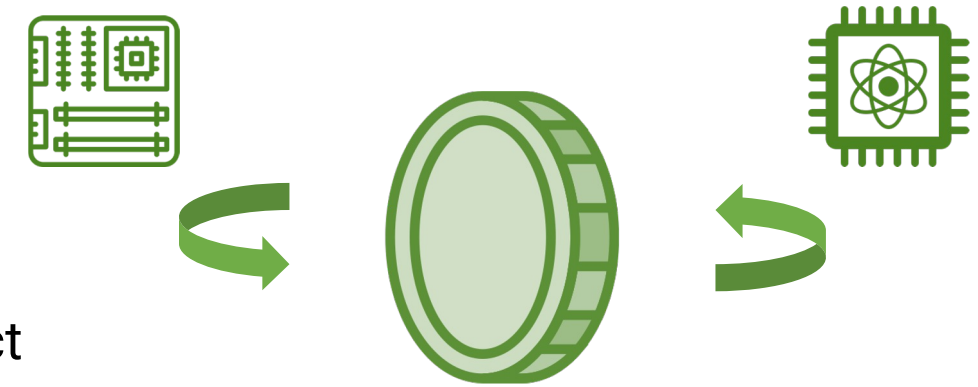
3. Third, the tutorial will **present designs for securing the cloud-based NISQ quantum computers** from the security attacks

4. Fourth, this tutorial will introduce **early ideas for Fault-Tolerant Quantum Computing (FTQC) security**

Tutorial on Security of Quantum Computing Systems

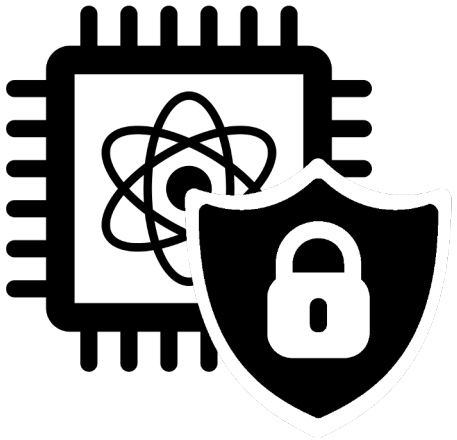
- What is not covered in the tutorial:
  - **Not covering Quantum Key Distribution and Quantum Networking** – a secure communication method leveraging properties of quantum mechanics
  - **Not covering Quantum Blind Computation** – algorithms for execution of quantum computation using one or more remote, untrusted quantum servers
  - **Not covering Quantum Cryptography** – algorithms exploiting quantum mechanical properties to perform cryptographic tasks
  - **Not covering Post-Quantum Cryptography** – cryptographic algorithms that are thought to be secure against a cryptanalytic attack by a quantum computer

Security + Quantum Computing: More than Post-Quantum Cryptography



Post-Quantum Cryptography: protect classical computers from Quantum Computer attacks

Quantum Computer Cybersecurity: protect Quantum Computers themselves from attacks



Tutorial on Security of Quantum Computing Systems

Tutorial Logistics

Computer Architecture and Security Lab (CASLAB)

Northwestern University

Tutorial Presenter

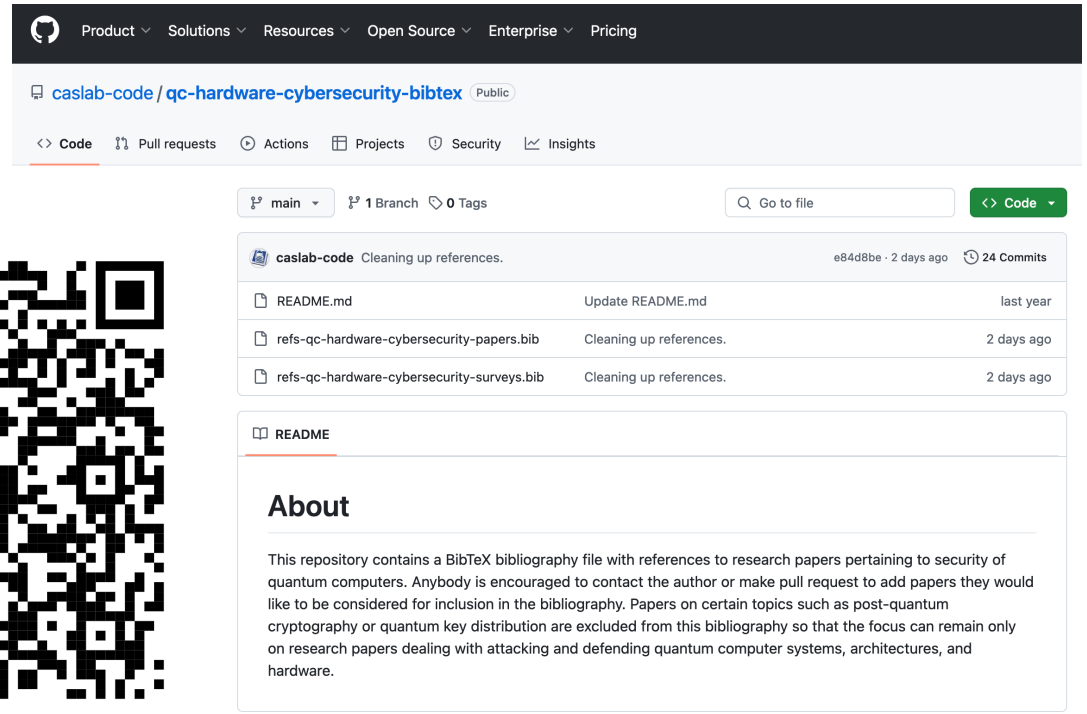
- Tutorial presenter:

- **Prof. Jakub Szefer**
  Computer Architecture and Security Lab (CASLAB)
  Northwestern University

https://caslab.io/jakub

Tutorial Resources

- Tutorial resources:
  - **Quantum Computer Hardware Cybersecurity BibTeX**
    - https://github.com/caslab-code/qc-hardware-cybersecurity-bibtex
    - A bibliography file containing references to Quantum Computer Hardware Cybersecurity research papers
    - Papers from various research groups, superset of work presented at this tutorial
    - Regularly updated



Tutorial Organization

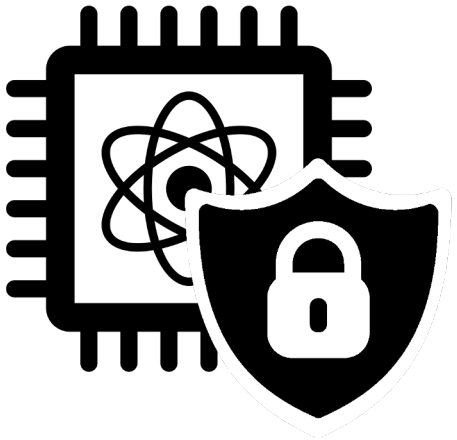
10:00am – 11:30am – Tutorial Part 1

- Motivation: Need for Securing Quantum Computers
- Brief Introduction to Classical Security Topics
- Overview of Threats to Quantum Computing Systems
- Fault Injection and Classification for NISQ Systems

11:30am – 1:00pm – Lunch

1:00pm – 2:30pm – Tutorial Part 2

- Side Channels in NISQ Systems
- Trusted Execution Environments for NISQ Systems
- Fault-Tolerant Quantum Computing (FTQC) Security



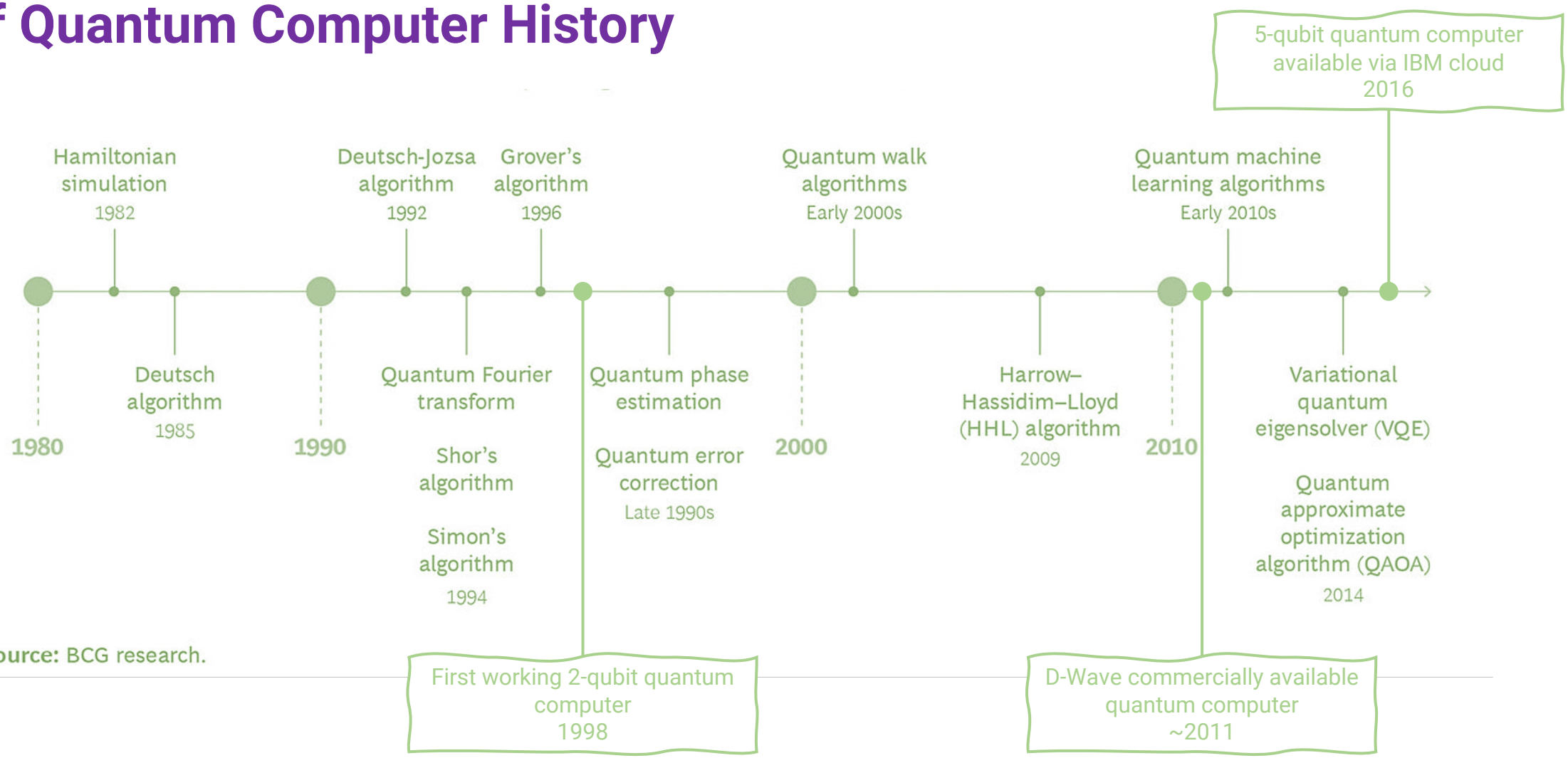
Tutorial on Security of Quantum Computing Systems

Motivation: Need for Securing Quantum Computers

Computer Architecture and Security Lab (CASLAB)

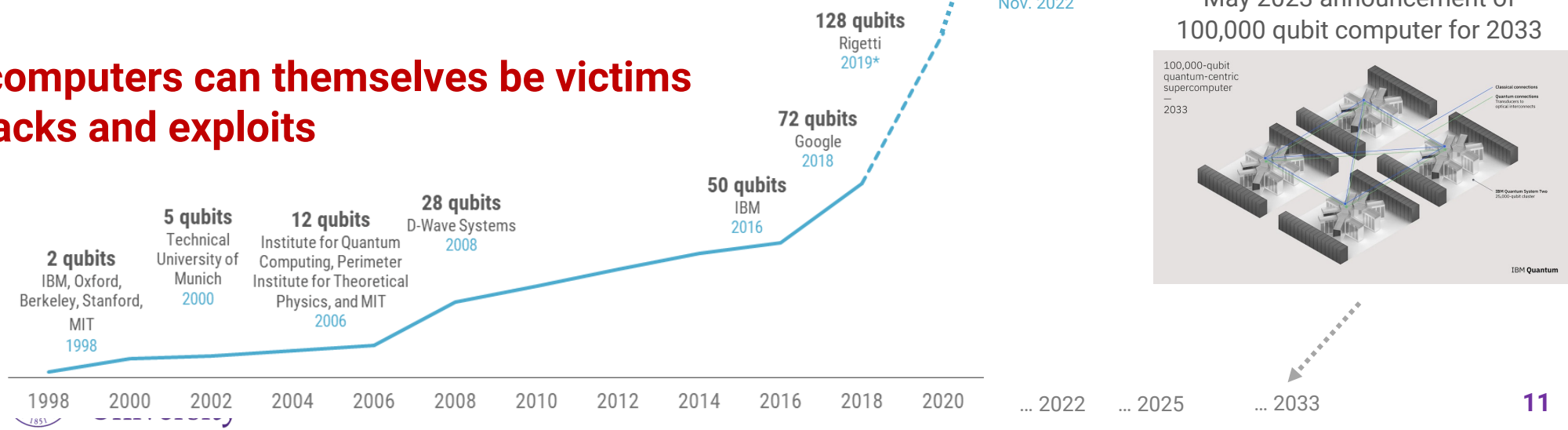
Northwestern University

Brief Quantum Computer History



Quantum Computing Today

- The field of quantum computing is undergoing rapid development
- Number of quantum bits (qubits) grows rapidly in recent years
- Quantum computers promise to generate novel results in, e.g., chemistry, materials research, or medicine
- They could be also used to attack classical cryptographic algorisms such as RSA
- **But quantum computers can themselves be victims of security attacks and exploits**



IBM Quantum

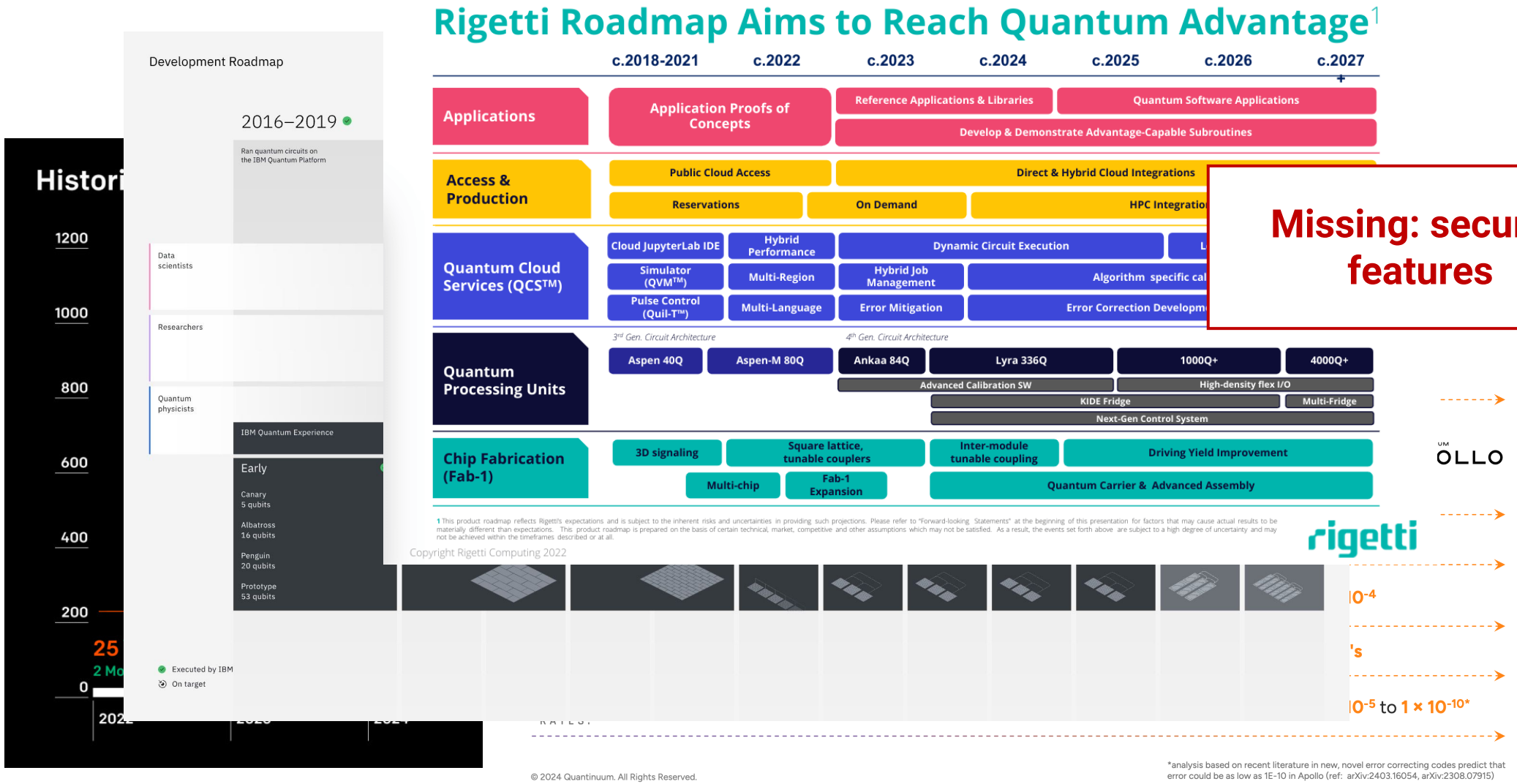
Cloud Providers

Amazon Braket

Microsoft Azure

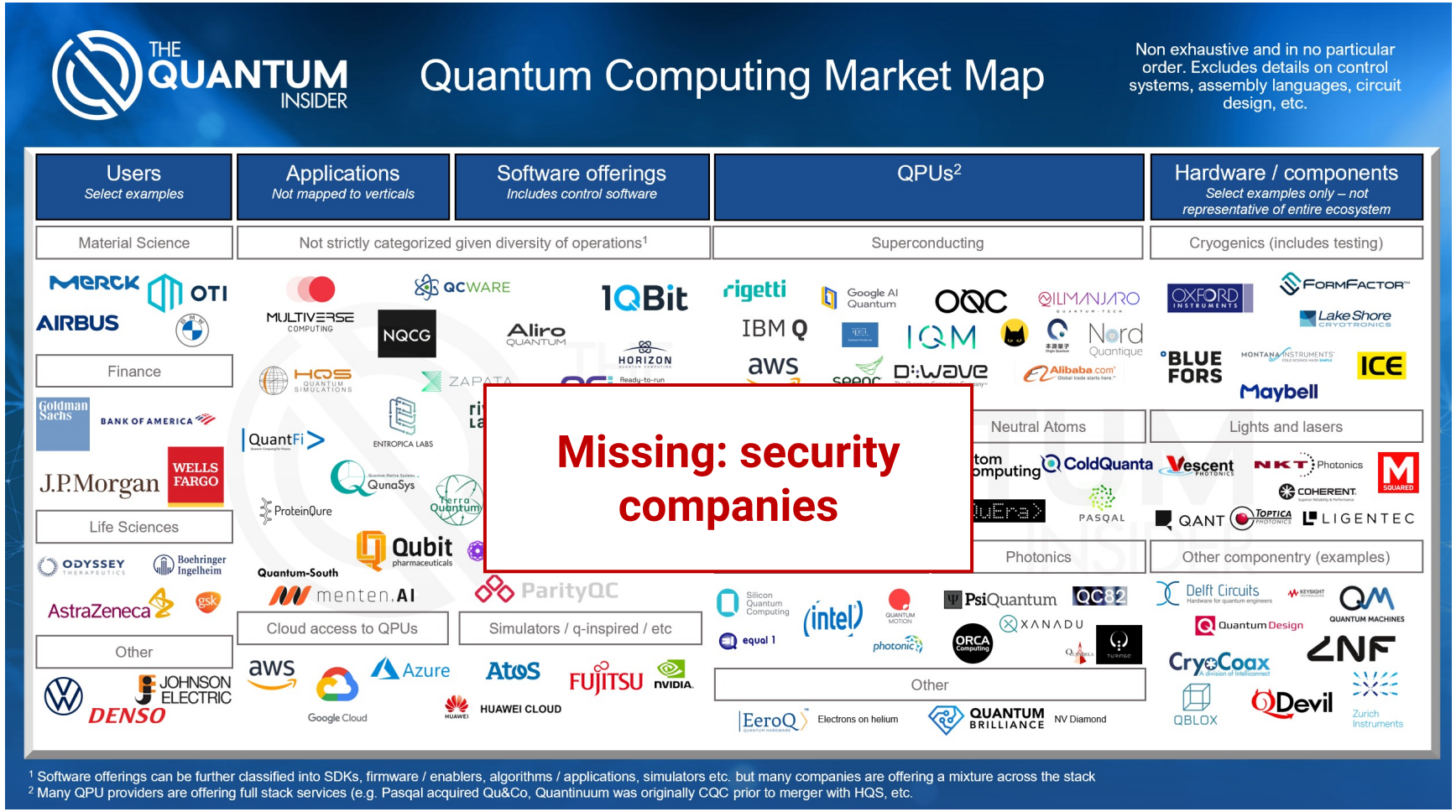
May 2023 announcement of 100,000 qubit computer for 2033

Roadmaps For Quantum Computing



Missing: security features

Quantum Computing Ecosystem



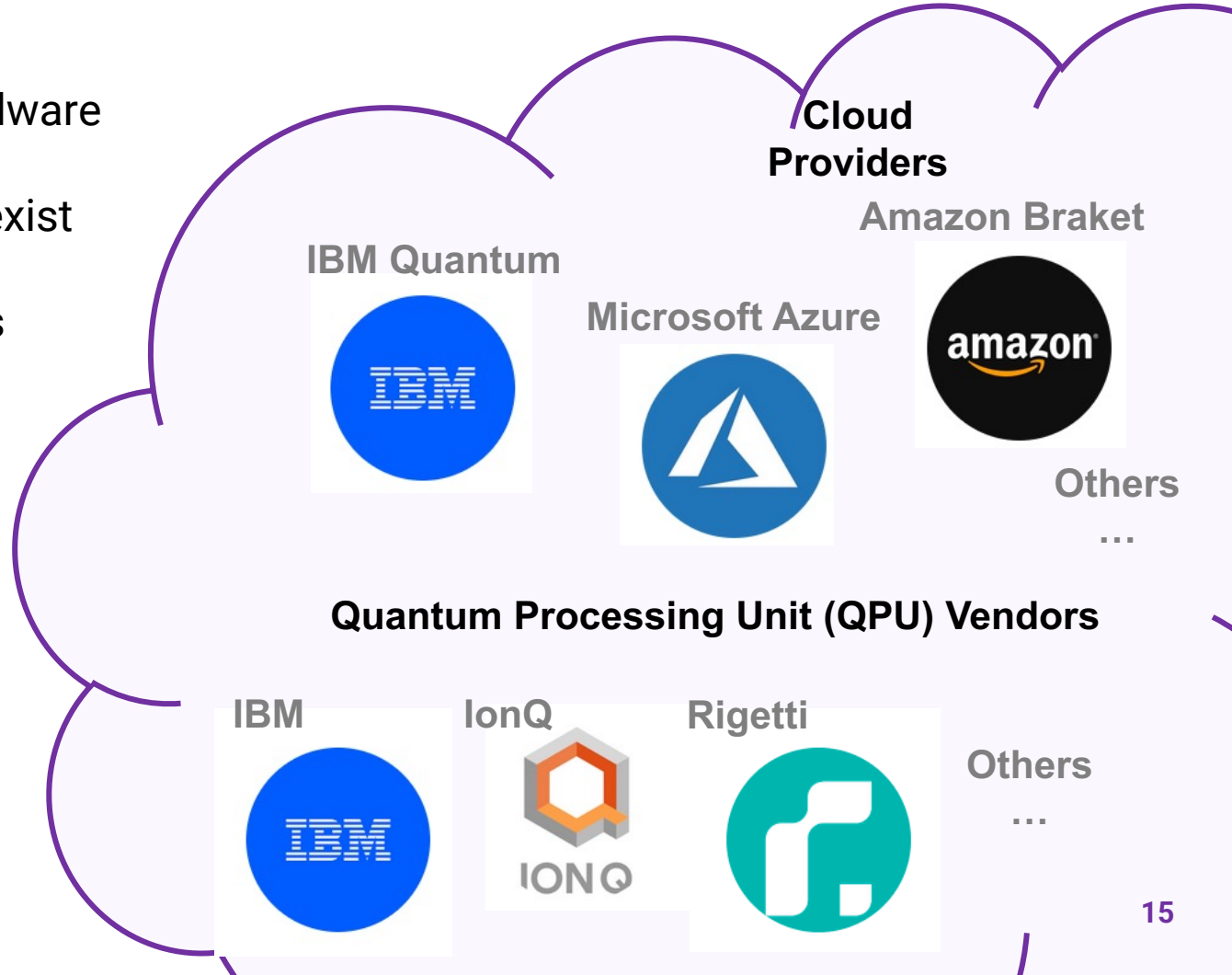
Motivation for Study of Security of Quantum Computing

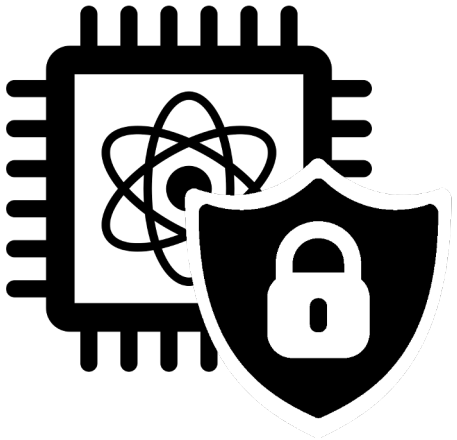
Adding **Security** gives **economic and social benefits** of **protecting private data** and **intellectual property** processed by **quantum computers**. It is also a building block for enabling **privacy** of **quantum data** and **computation**.

Cloud-based Access Further Motivates Security Research

- Quantum Computers are rapidly being deployed as cloud-based accelerators today
  - Size is still very limited ~127 qubits
  - But can relatively easily experiment with hardware through cloud-based access
  - Free and paid cloud-based services already exist
  - Many companies are pushing for Quantum Computer as a Service (QCaaS) deployments

- But there are no security considerations in QCaaS today
  - Possibly malicious users can now access quantum computing hardware remotely
  - Can attack other users, or try to reverse engineer the infrastructure





Tutorial on Security of Quantum Computing Systems

Brief Introduction to Classical Security Topics

Computer Architecture and Security Lab (CASLAB)

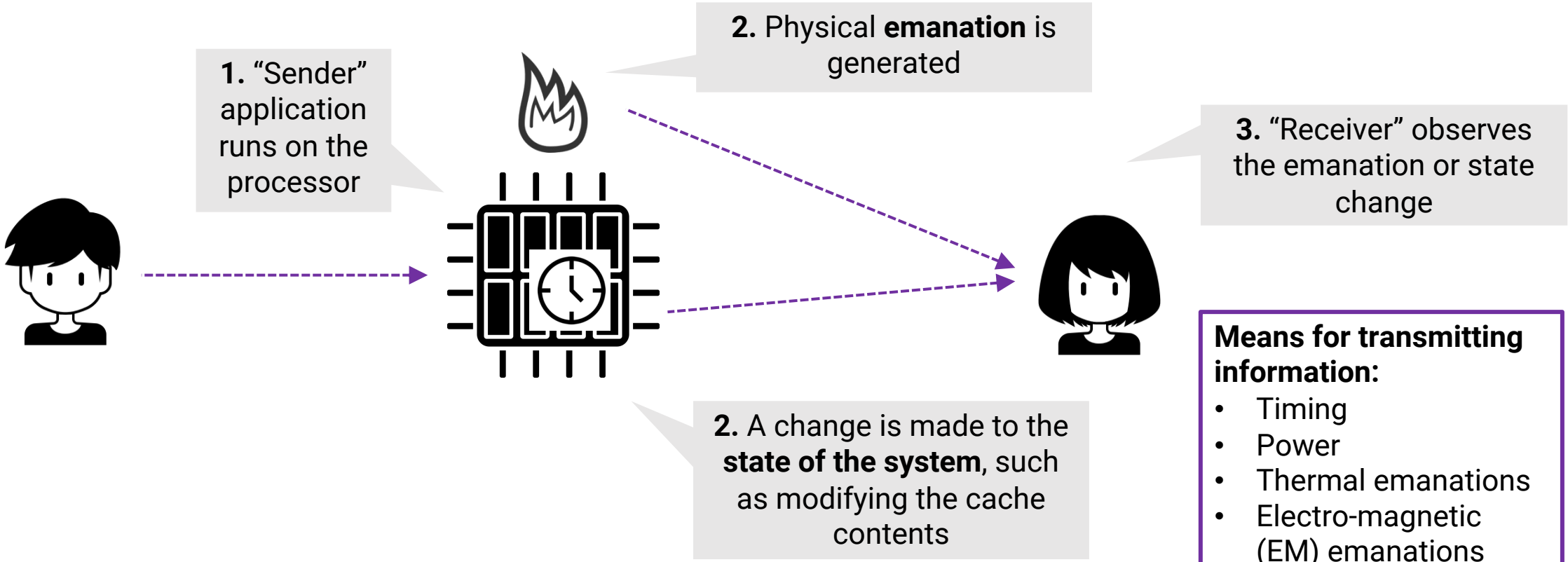
Northwestern University

Confidentiality, Integrity, Availability

- **Confidentiality** is the property, that information is not made available or disclosed to unauthorized individuals, entities, or processes
- **Integrity** is the property, that information has not been manipulated by unauthorized individuals, entities or processes
- **Availability** is the property, that information can be accessed when required by authorized individuals, entities or processes
- Security threats and attacks can be described by which property they violate
  - E.g. side-channel attack leaks information and violates confidentiality
  - E.g. fault-injection attack can modify information and violates integrity
- Often “information” refers to code or data or both

Covert Channels *(Classical Computing Examples)*

- A **covert channel** is an intentional communication between a sender and a receiver via a medium not designed to be a communication channel.

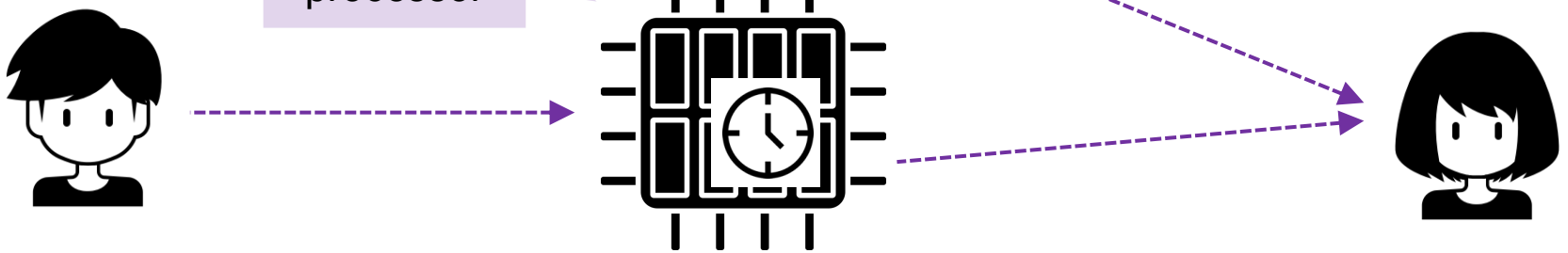


Means for transmitting information:

- Timing
- Power
- Thermal emanations
- Electro-magnetic (EM) emanations
- Acoustic emanations

Side Channels *(Classical Computing Examples)*

- In a **side channel**, the "sender" in an unsuspecting victim and the "receiver" is the attacker.



1. "**Victim**" application runs on the processor

2. Physical **emanation** is generated

3. "**Attacker**" observes the emanation or state change

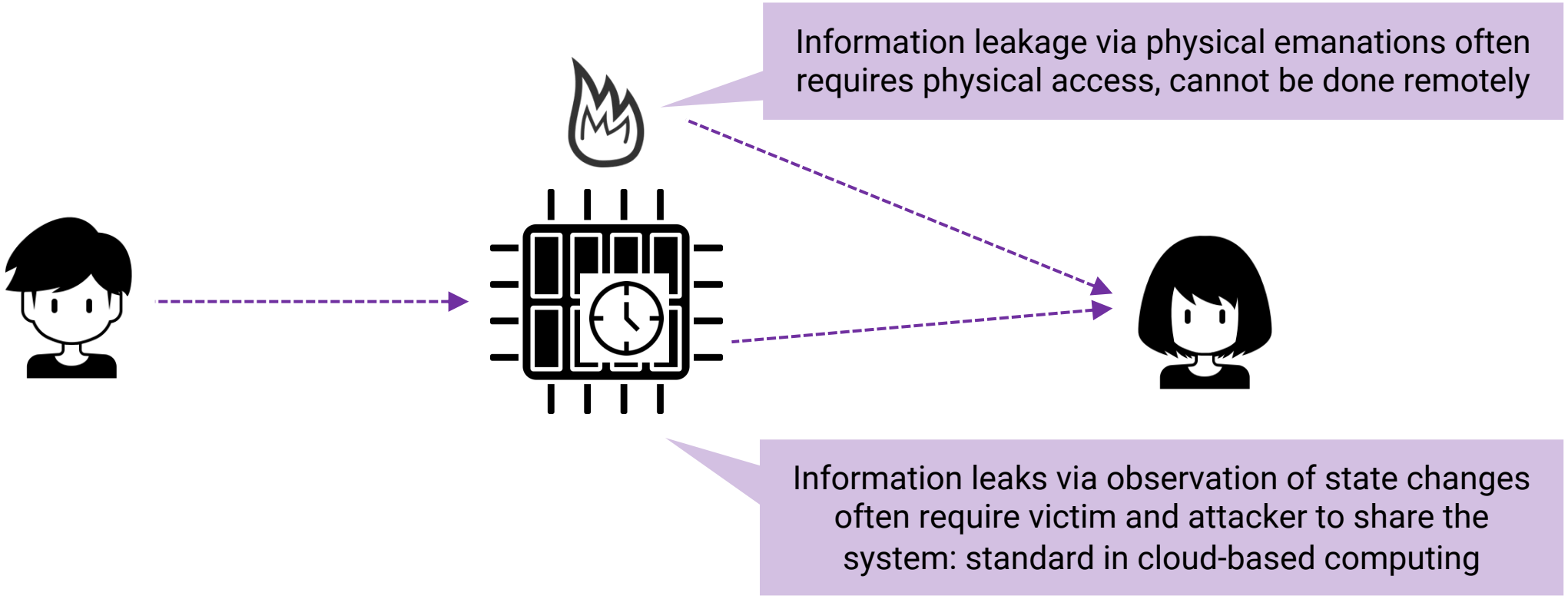
2. A change is made to the **state of the system**, such as modifying the cache contents

Means for transmitting information:

- Timing
- Power
- Thermal emanations
- Electro-magnetic (EM) emanations
- Acoustic emanations

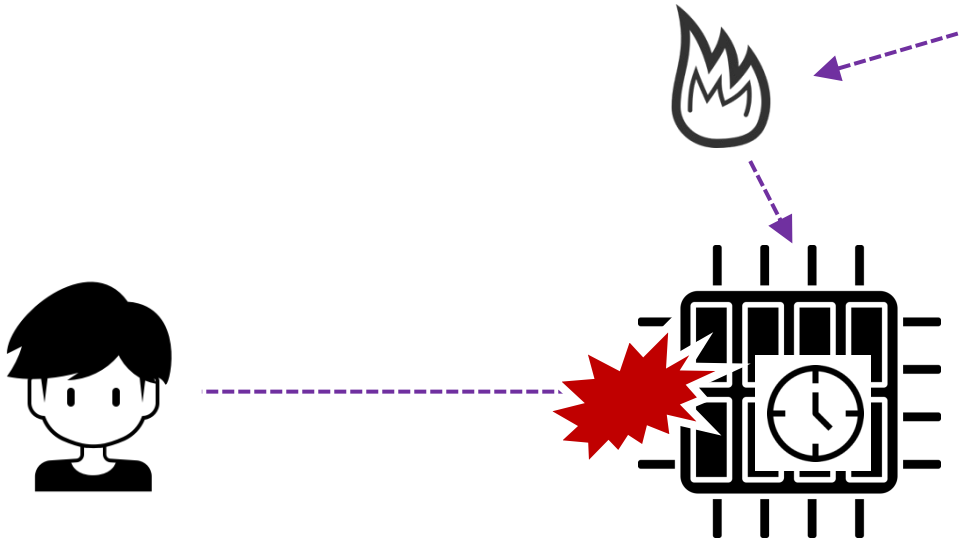
Information Leakage *(Classical Computing Examples)*

- **Information leakage occurs through side or covert channels,** either with or without physical access; many interesting information leaks are remote

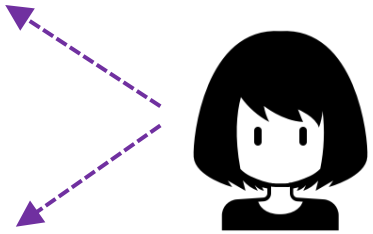


Fault Injection *(Classical Computing Examples)*

- **Fault injection attacks involve malicious, often temporary, modification of the system** to cause a change in behavior, i.e. a fault



Fault injection with physical access:
- Clock glitching – alter clock timing
- Voltage glitching – momentarily drop supply voltage
- EM radiation – generate high-intensity EM pulse
- Optical fault injection – laser fault injection



Fault injection without physical access:
- RowHammer – flip bits in DRAM
- Dynamic Voltage and Frequency Scaling – alter voltages

Reverse Engineering *(Classical Computing Examples)*

- **Reverse engineering** involves observing the system operation or physically probing the system to learn how it is built or how it operates

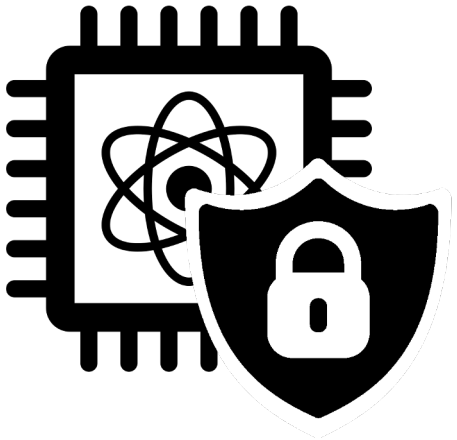
Reverse engineering:
- Remotely – observe contention or resource sharing
- Physically – probe or even destructively examine the system

Review: Confidentiality, Integrity, Availability

- **Confidentiality** is the property, that information is not made available or disclosed to unauthorized individuals, entities, or processes
- **Integrity** is the property, that information has not been manipulated by unauthorized individuals, entities or processes
- **Availability** is the property, that information can be accessed when required by authorized individuals, entities or processes

Review: Main Security Threats

- **Side Channels** can be used to bypass logical isolation mechanisms and access control to leak sensitive information (code or data)
- **Fault Injection** can be used to change the state of the system, from simple denial of service attacks to manipulation of the system to enter specific state or produce specific output
- **Reverse Engineering** can be used to learn how the system is built and to steal intellectual property or trade secrets



Tutorial on Security of Quantum Computing Systems

Overview of Threats to Quantum Computing Systems

Computer Architecture and Security Lab (CASLAB)

Northwestern University

Qubits and Quantum States

In a two-state quantum-mechanical system, the general form of a qubit (a quantum bit) state can be represented by:

Qubit =
Quantum Bit =
Quantum Register

$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

- α and β are complex numbers that specify the probability amplitudes of the corresponding states

Notes:

- $|\alpha|^2$ gives the probability that you will find the qubit in 0 state; and $|\beta|^2$ gives the probability that you will find the qubit in the 1 state
- α and β are complex probability amplitudes satisfying equation $|\alpha|^2 + |\beta|^2 = 1$

Multi-Qubit Quantum States

Quantum states can be composed of multiple qubits as well:

- A state of a 2-qubit system can be described as follows:

2 Qubit System =
2 Qubit Register

$$|\psi\rangle = \alpha_{00}|00\rangle + \alpha_{01}|01\rangle + \alpha_{10}|10\rangle + \alpha_{11}|11\rangle$$

- The coefficients still need to satisfy the requirement that squares of their amplitudes add up to a sum of 1: $|\alpha_{00}|^2 + |\alpha_{01}|^2 + |\alpha_{10}|^2 + |\alpha_{11}|^2 = 1$

- When doing a measurement, we obtain a bitstring $|00\rangle$ with probability $|\alpha_{00}|^2$, $|01\rangle$ with probability $|\alpha_{01}|^2$, etc.

Qubits, Qutrits, and More

- A **qutrit** is a quantum version of a trit (a classical 3-level logic unit) and it lives in a three-dimensional Hilbert space, meaning its state is a superposition of 3 basis states:

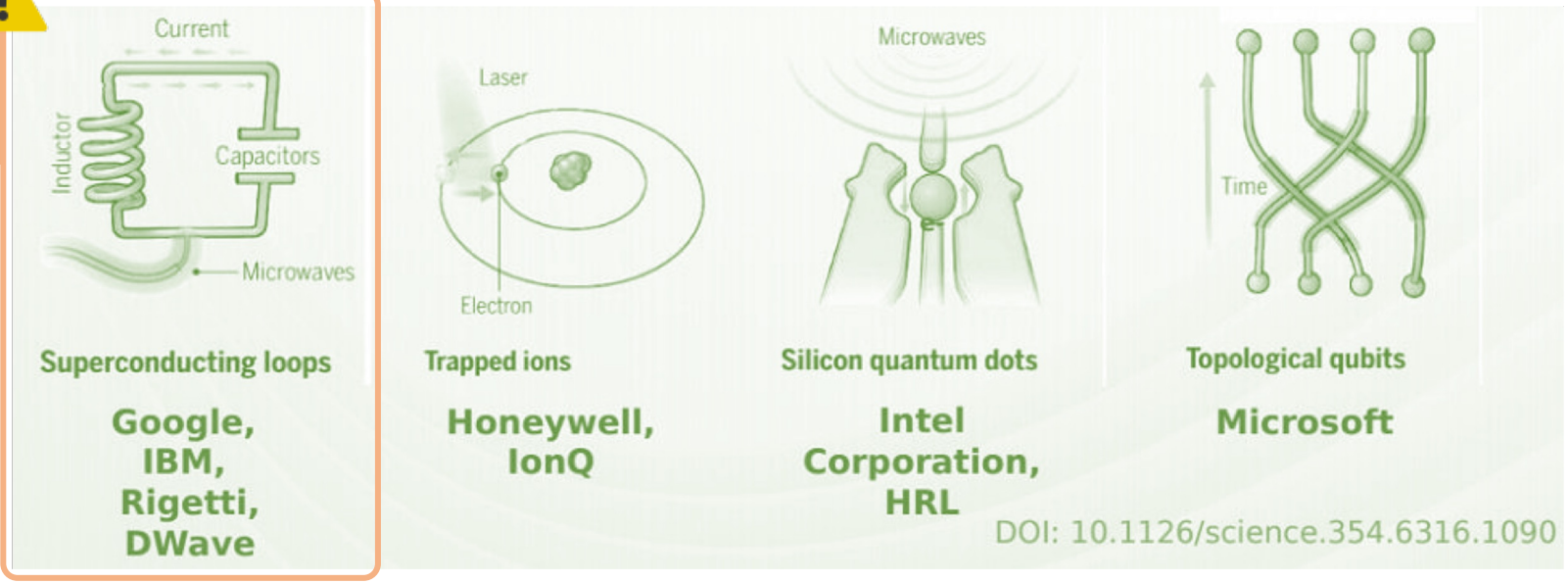
$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle + \gamma|2\rangle$$

- A **qumode** is an extension of the concept of qubits to continuous-variable quantum systems. While qubits are discrete and take values from a two-level system $|0\rangle$ and $|1\rangle$, qumodes are used to describe quantum systems with an infinite number of possible states, typically in continuous Hilbert spaces.

Most of the discussion in the tutorial will focus on qubits.

Qubit Types

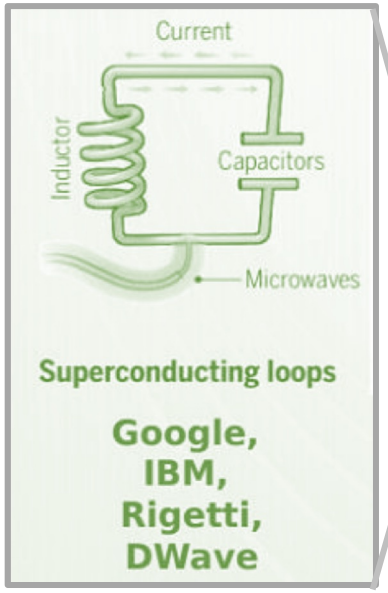
Tutorial will focus on superconducting qubits.



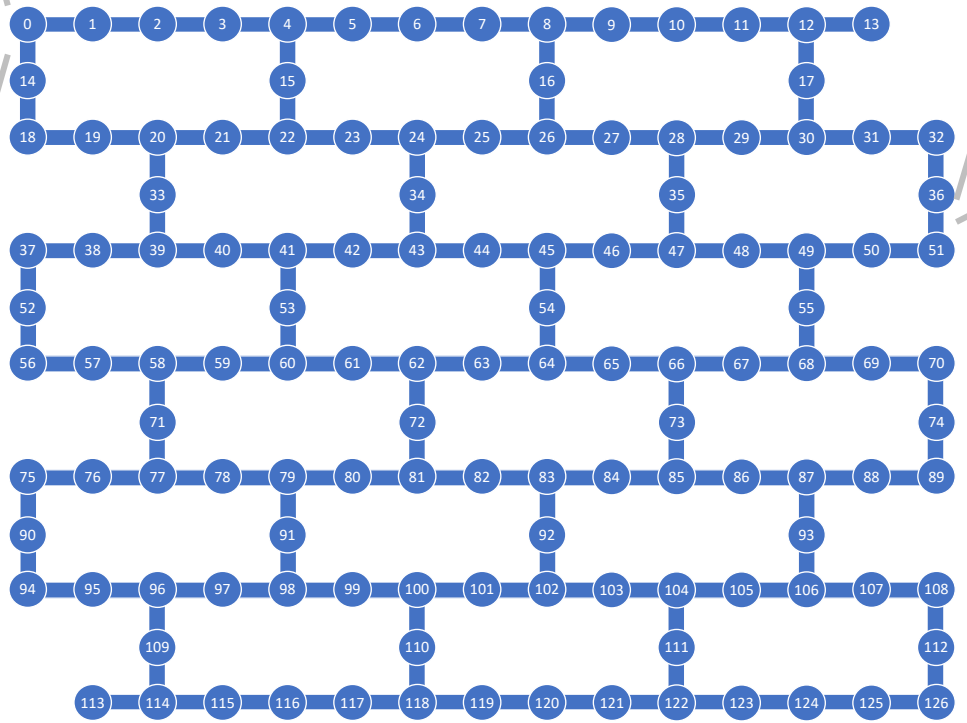
Many ways to realize a qubit:

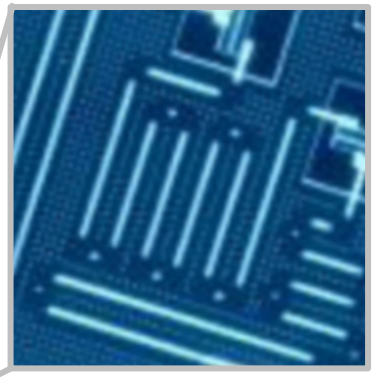
- Nuclear spin in NMR (Nuclear magnetic resonance) system: ↑ = $|0\rangle$, ↓ = $|1\rangle$
- Photons in a cavity: 0 photon = $|0\rangle$, 1 photon = $|1\rangle$
- Energy states of an atom: ground state $|0\rangle$, excited state $|1\rangle$
- Energy levels of transmon circuit: ground state $|0\rangle$, first excited state $|1\rangle$

Superconducting Qubit Quantum Computer Topology



A transmon quit, used to store qubit state.





A coupler, used for 2-qubit gate operations

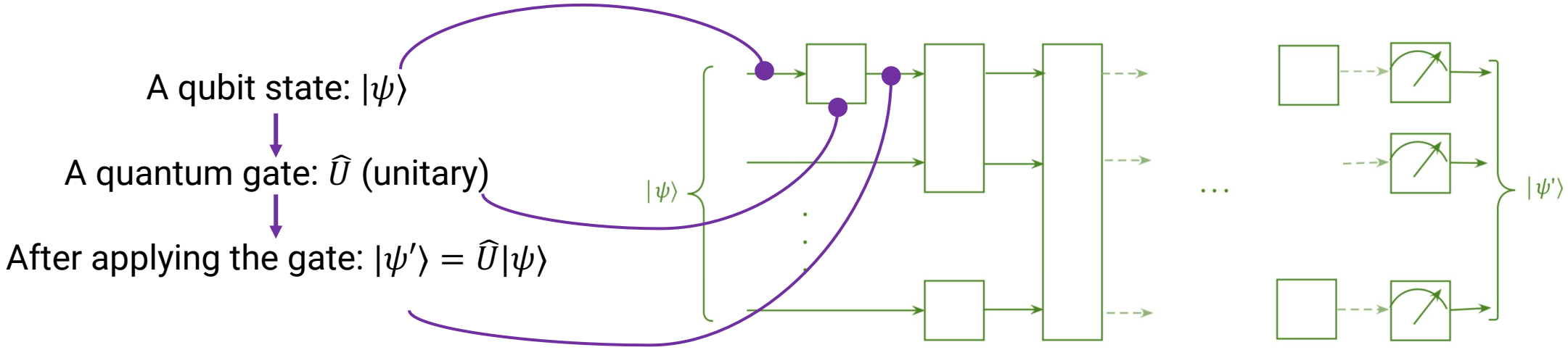
$$|\psi\rangle = \alpha_{0\ldots00}|0\ldots00\rangle + \alpha_{0\ldots01}|001\rangle + \ \ldots$$

Topology of IBM Quantum Eagle R3 Processor with topology for 127 qubits.

Gate Operations with Quantum Gates

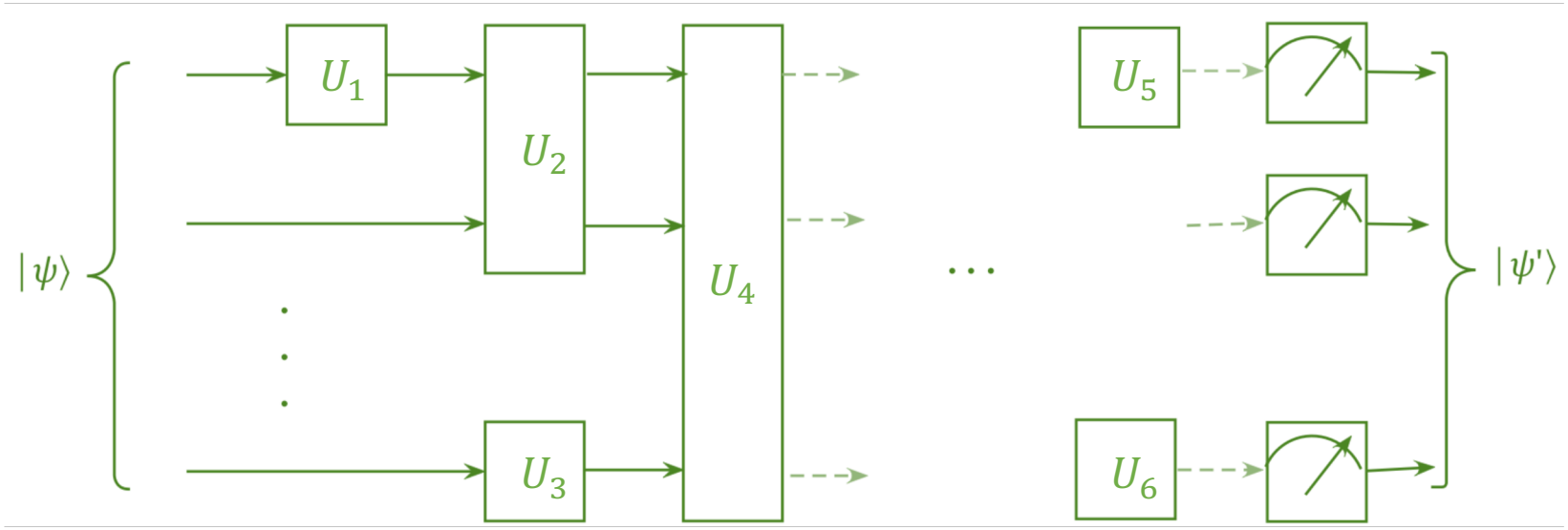
- In quantum computing, **quantum gates** such as Hadamard, Pauli-X, or CNOT, are all represented by **unitary operations**

- These gates effectively apply $\hat{U}$ transformation to qubits, evolving their state over time in discrete steps:



Quantum Computing with Quantum Gates

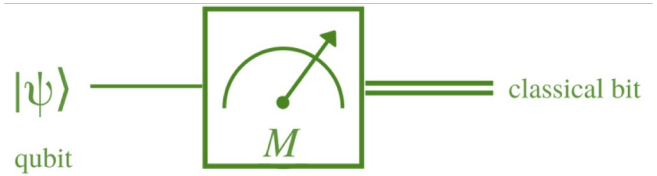
- On a quantum computer, programs are executed by unitary evolution (quantum gates) of an input that is given by the state of the system, $|\psi\rangle$, to generate the next state, $|\psi'\rangle$.



- Since all unitary operators are invertible, we can always reverse or “uncompute” a computation on a quantum computer.

Measurement and Quantum Computation Outputs

- If a quantum system were perfectly isolated, it would maintain coherence indefinitely, but it would be impossible to manipulate or read it out.



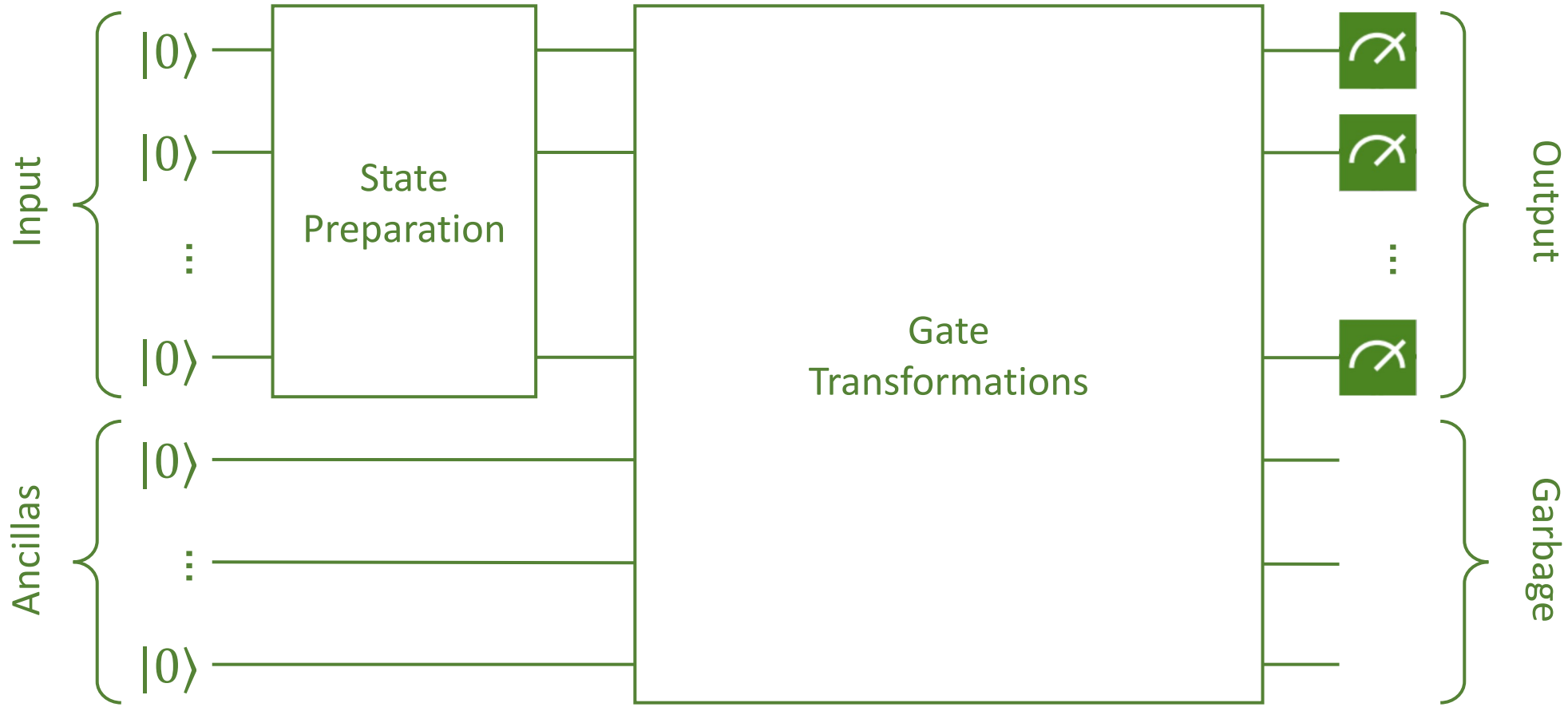
- A quantum measurement is a destructive process.

- When a quantum system is measured, the wave function $|\psi\rangle$ collapses to a new state according to a probabilistic rules.

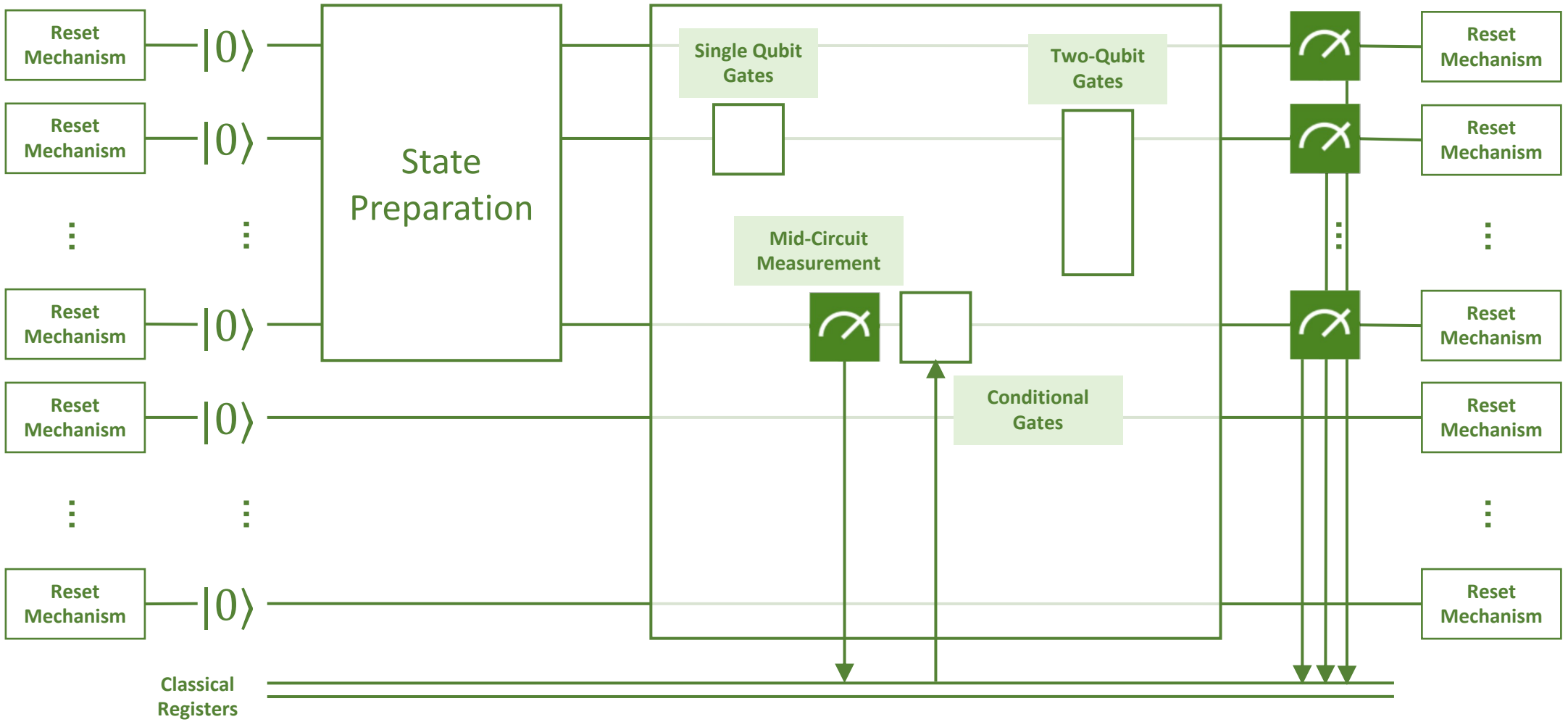
- If $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ after measurement, either $|\psi\rangle = |0\rangle$ or $|\psi\rangle = \alpha|1\rangle$, and these possibilities occur with probabilities of $|\alpha|^2$ and $|\beta|^2$ with $|\alpha|^2 + |\beta|^2 = 1$.

- A quantum measurement never produces another quantum state $|\psi\rangle = \alpha'|0\rangle + \beta'|1\rangle$

Putting It Together: A Typical Quantum Circuit



A Typical Quantum Circuit with More Details



Imperfections, Noise and Errors in Quantum Circuits

Some sources of noise and errors in NISQ quantum computers:

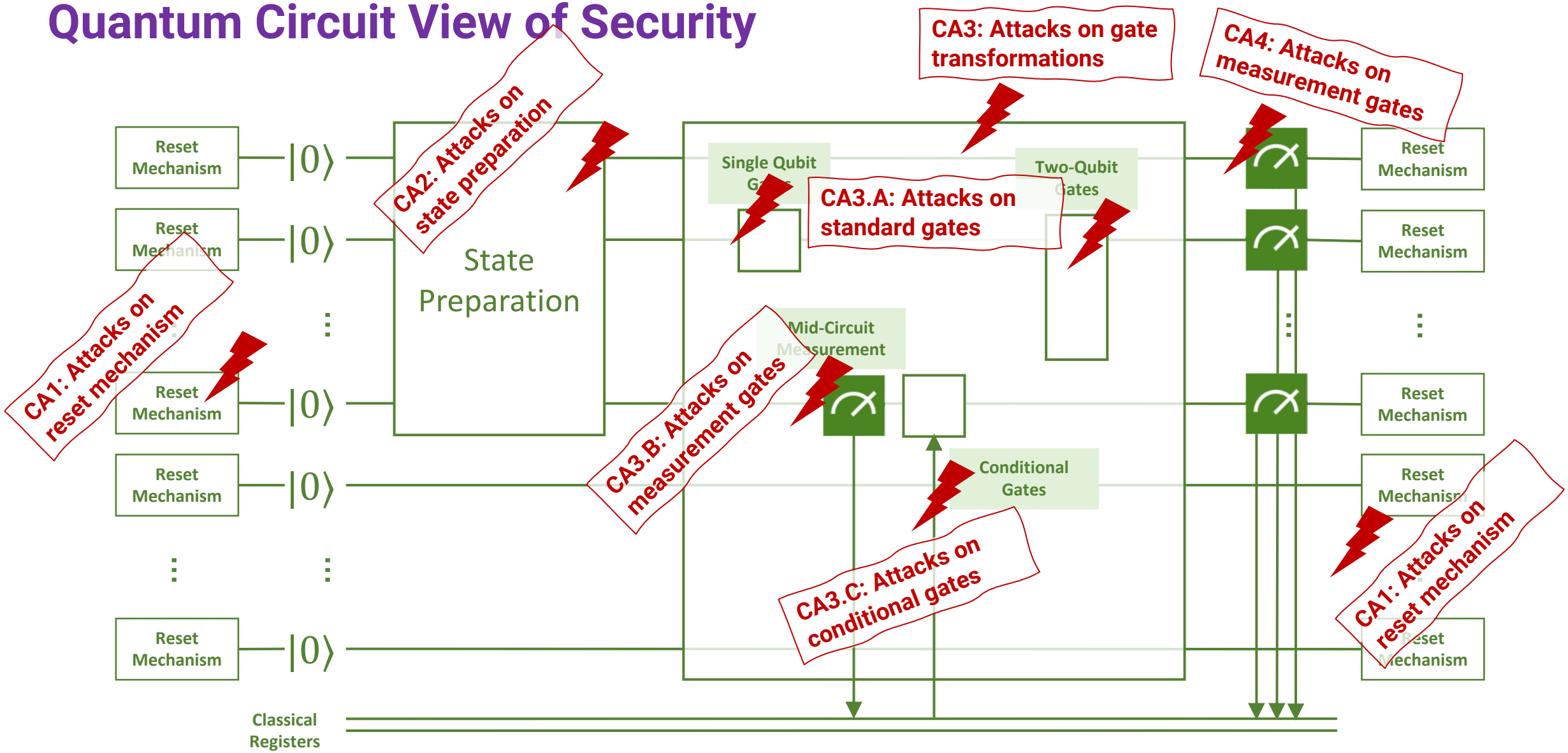
- **Imperfect gate operations** – variations in control pulses (timing, amplitude, phase) lead to gate operations no executing fully as expected
- **Crosstalk** – manipulating one qubit can affect another qubit
- **Leakage Errors** -- qubits can transition into states outside the logical $|0\rangle$ and $|1\rangle$ space, e.g., in multi-level systems like superconducting qubits
- **Readout errors** – Imperfect measurement processes cause incorrect detection of qubit states

Security Implications of Noise and Errors in Quantum Circuits

Attackers can leverage the noise and errors in today's NISQ quantum computers:

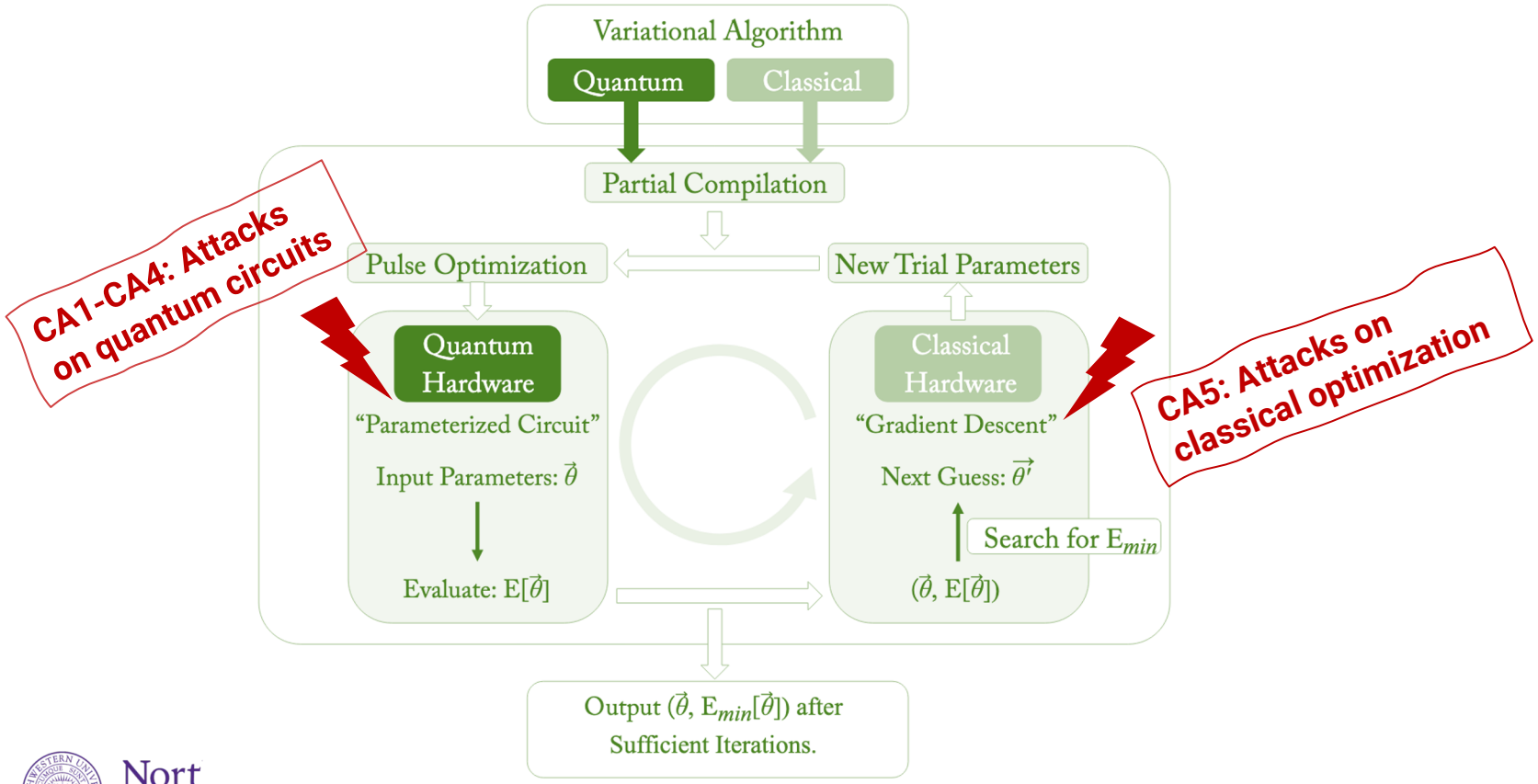
- **Imperfect gate operations** – side channels or fault injection
- **Crosstalk** – side channels or fault injection
- **Leakage Errors** – fault injection
- **Readout errors** – side channels or fault injection

Quantum Circuit View of Security



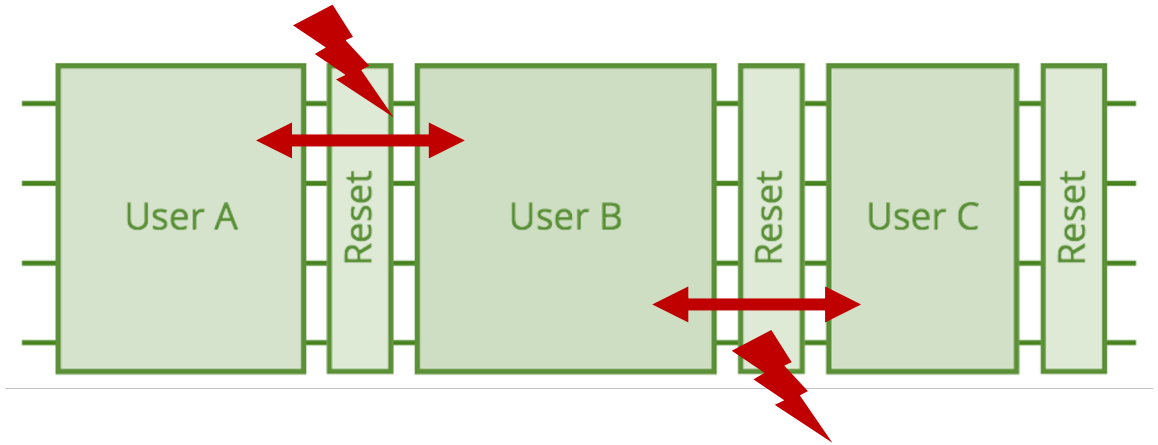
Expanded Quantum Circuit View of Security: Variational Circuits

Variational quantum circuits, e.g., QAOA, VQLS, etc., are hybrid quantum-classical models that use parameterized quantum gates optimized via classical algorithms:

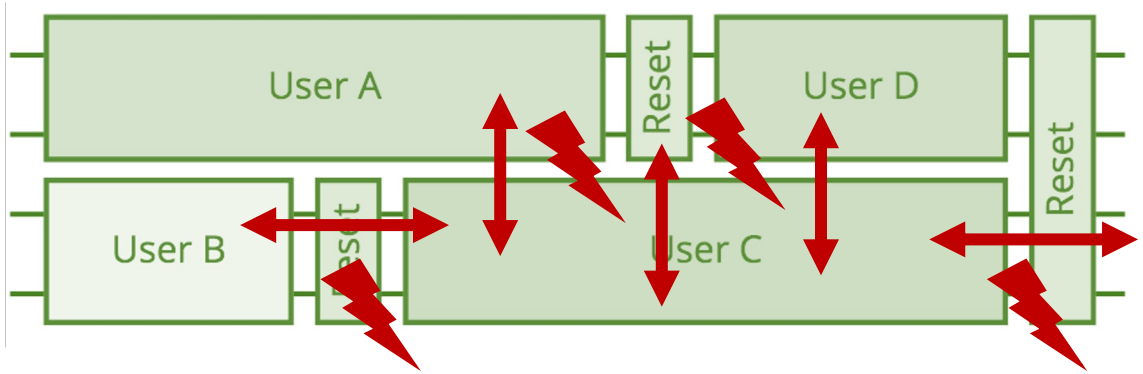


Quantum Circuits in Single- and Multi-Tenant Quantum Computers

- It is possible to share the quantum computing hardware in single- or multi-tenant ways leading to possible remote attacks among users:



Single-tenant

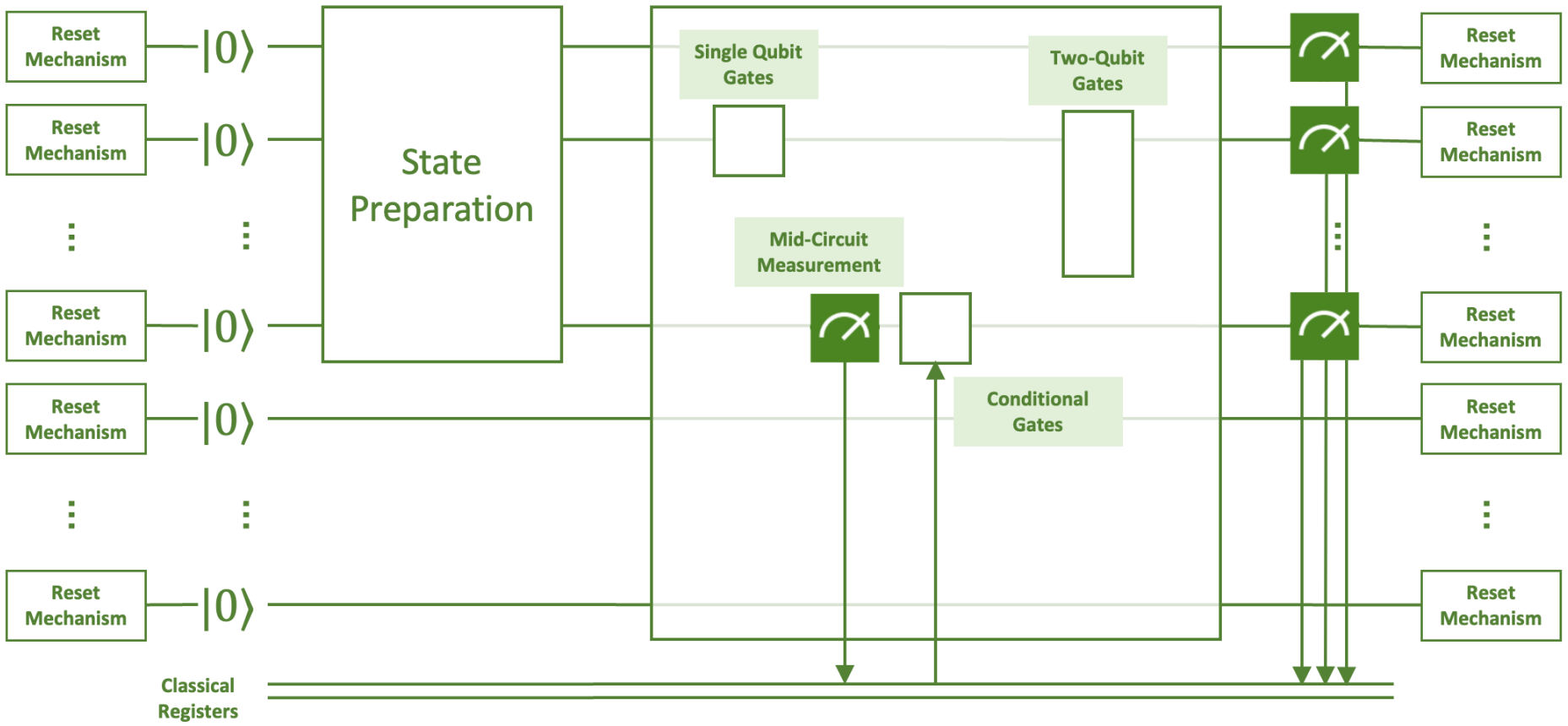


Multi-tenant

No quantum computer provider publicly uses multi-tenant configuration as of 2025

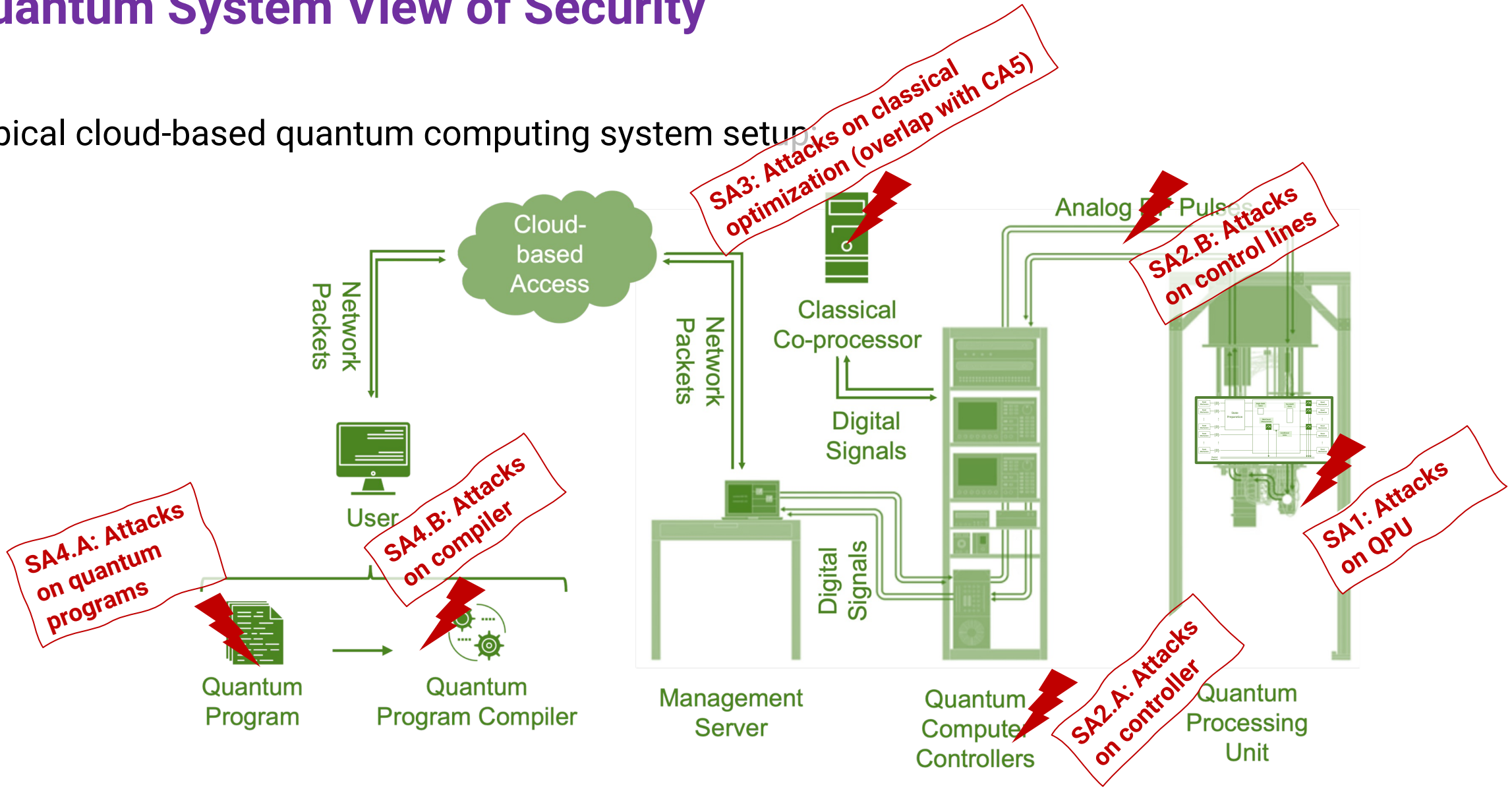
Quantum System View of Security

Typical cloud-based quantum computing system setup:



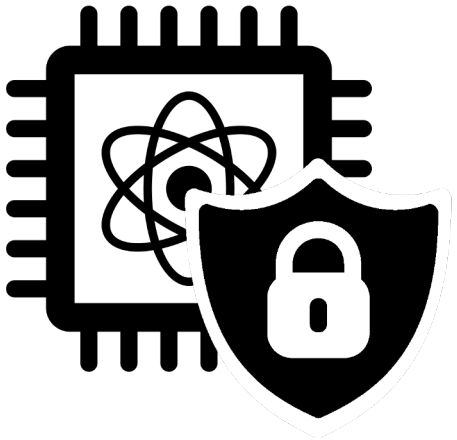
Quantum System View of Security

Typical cloud-based quantum computing system setup:



Summary: Quantum Computing Systems and Security

- Quantum computers are still “computers” making them vulnerable to various security attacks
- **Pros of quantum computers:**
  - Cannot clone quantum states, cannot directly “steal” information like in classical computers
- **Cons of quantum computers:**
  - NISQ computers are prone to noise and errors
    - Can exploit these for side channels and fault injection
  - Require extensive control equipment which itself can be vulnerable to attacks
    - Control equipment not well studied from security perspective before
  - Large physical size makes physical size and probing easier
    - Although most computers are so far highly guarded in data centers
  - Currently lack of quantum memory and networking
    - Data is hard-coded into code
  - Rapidly being deployed in the cloud, without security considerations
    - Easy to run any code without security checks



Tutorial on Security of Quantum Computing Systems

Fault Injection and Classification for NISQ Systems

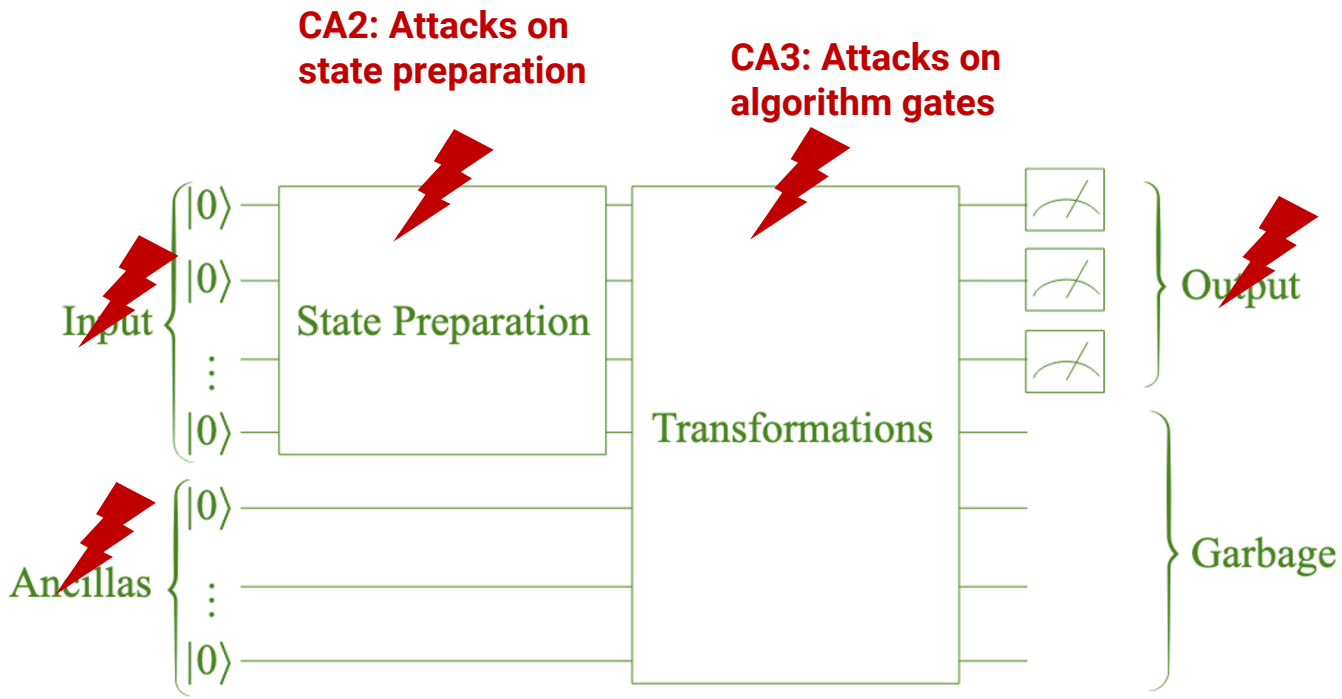
Computer Architecture and Security Lab (CASLAB)

Northwestern University

Fault Injection and Crosstalk Attacks

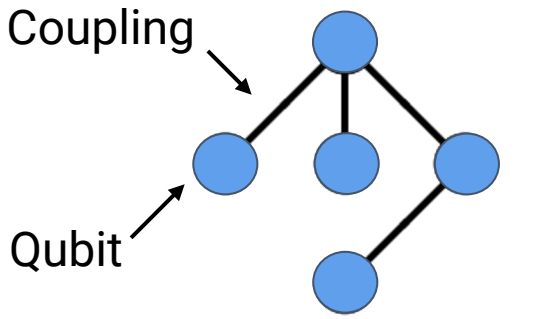
CA1.A: Attacks on initial input qubits states

CA1.B: Attacks on initial ancilla qubits states



Crosstalk Attacks Among Circuits

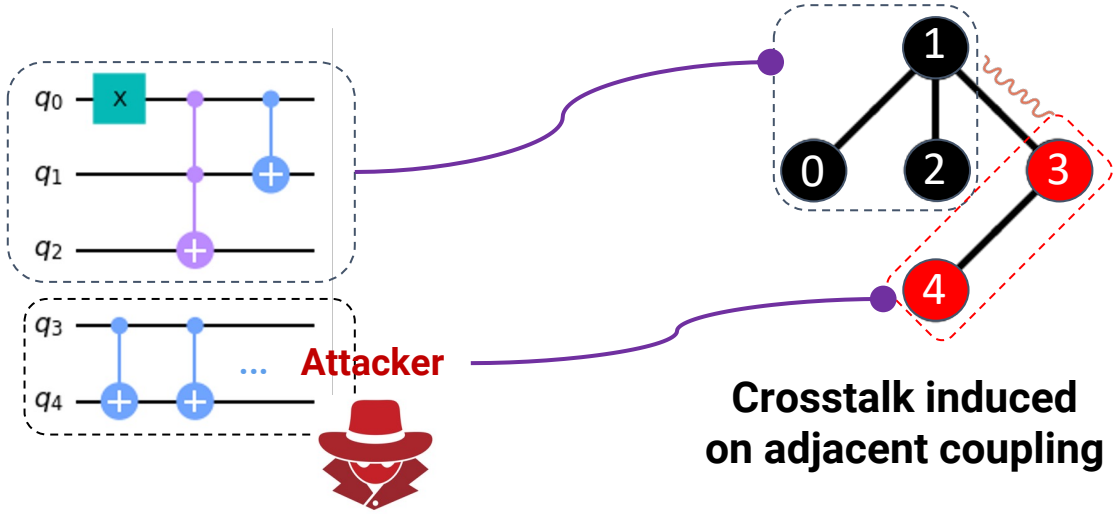
- Crosstalk is the effect of performing computation on one or more qubits unintentionally affecting one or more other qubits
- Based on how the Qubits are arranged, signal intended for one qubit may affect other qubits through crosstalk



Example QPU topology

Quantum Half Adder

Series of CNOT gates

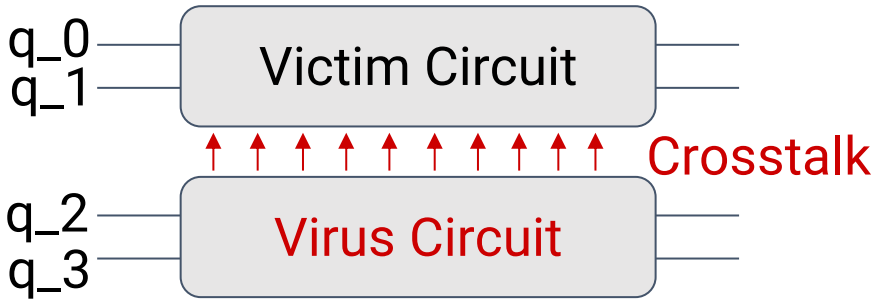


Crosstalk induced on adjacent coupling

Reference:
Deshpande, et al. (HOST 2022)

Typical Crosstalk Attack Setup

- Assume that victim and attacker share the same QPU

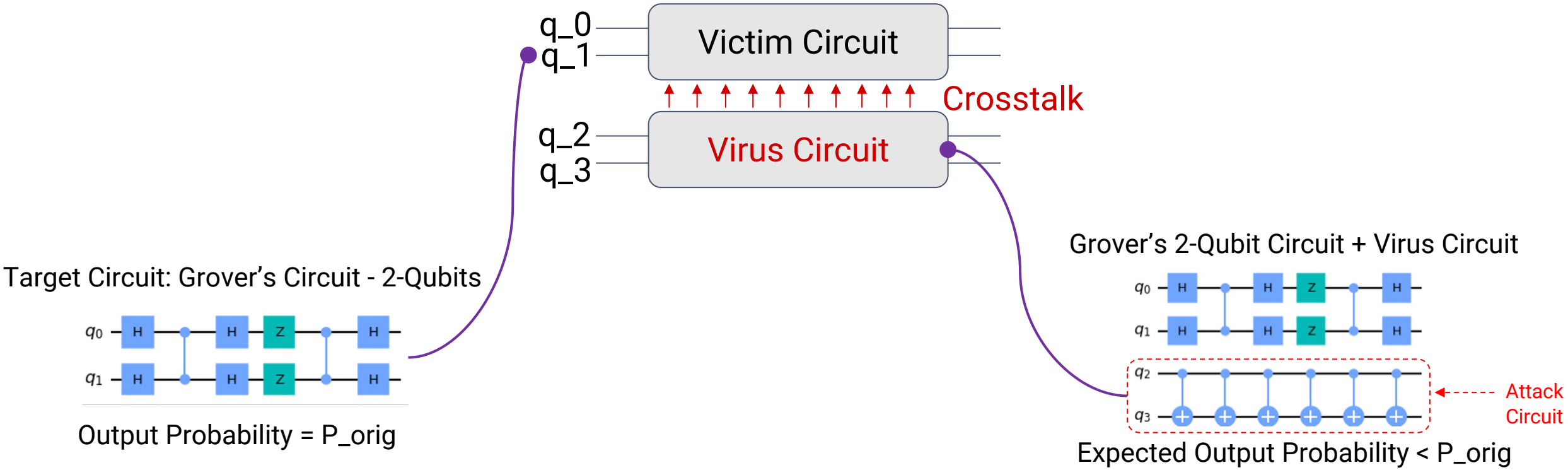


- Attacker circuit aims to reduce fidelity of the victim circuit

Reference:
Deshpande, et al. (HOST 2022)

Typical Crosstalk Attack Setup

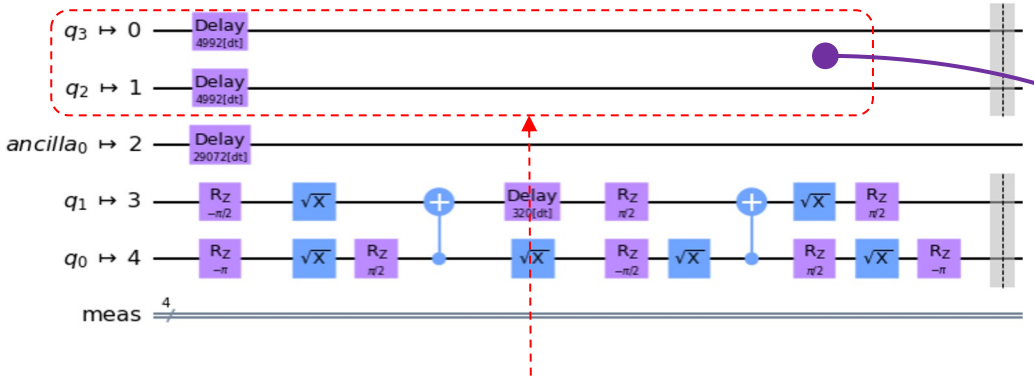
- Assume that victim and attacker share the same QPU



Reference:
Deshpande, et al. (HOST 2022)

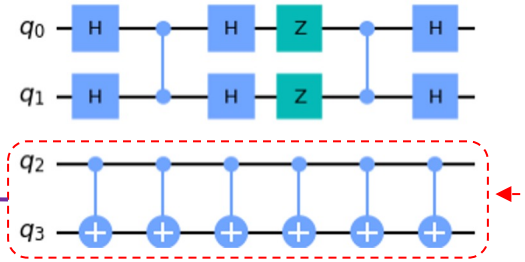
Attackers Need to Bypass Compiler Optimizations

- Compilers optimize away useless circuits
  - Circuit of all CNOT gates will be optimized away



Transpile function decomposes and removes the virus circuit

Grover's 2-Qubit Circuit + Virus Circuit



Attack Circuit

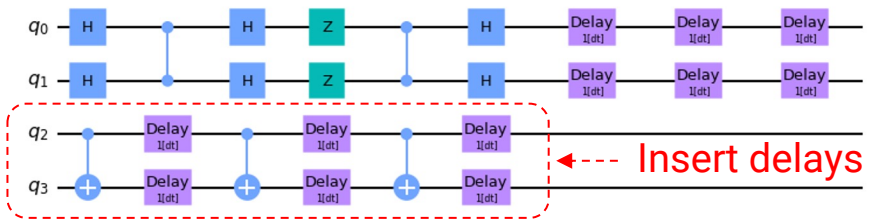
Expected Output Probability < P_orig

Reference:
Deshpande, et al. (HOST 2022)

Bypassing Compiler Optimizations

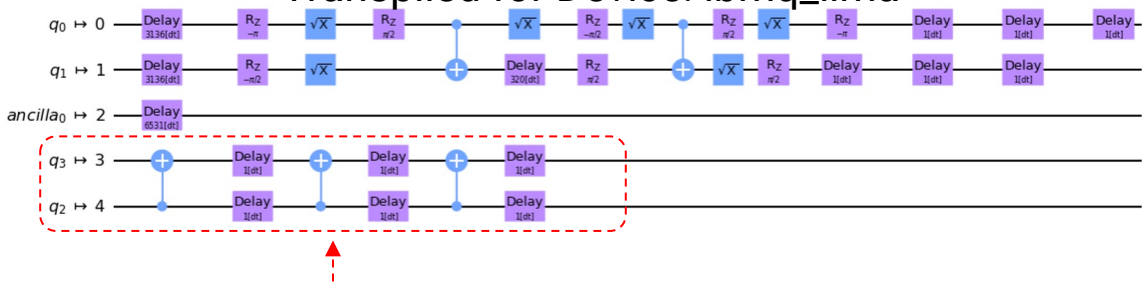
- Attack circuit with delays inserted

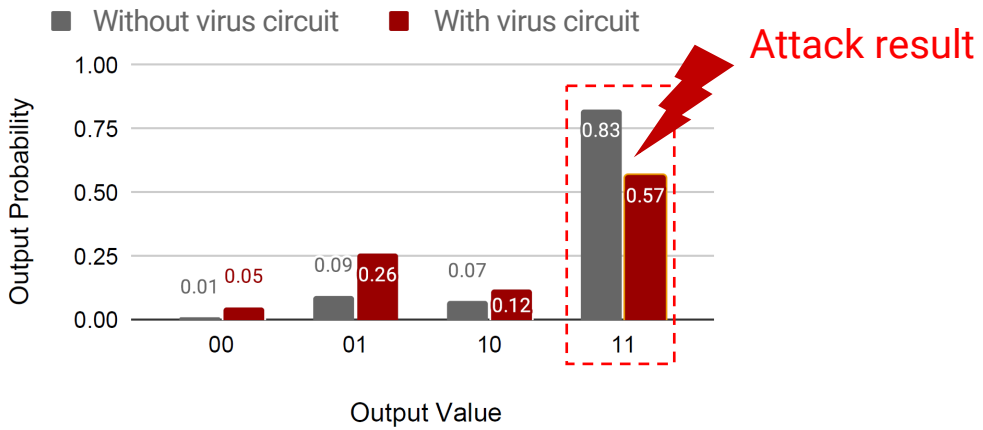
Grover's 2-Qubit Circuit + Proposal for Virus Circuit



Expected Output Probability < P_orig

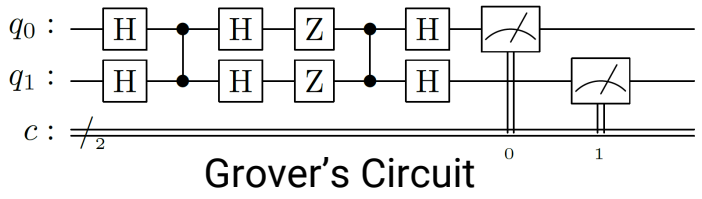
Grover's 2-Qubit Circuit + Virus Circuit
Transpiled for Device: ibmq_lima



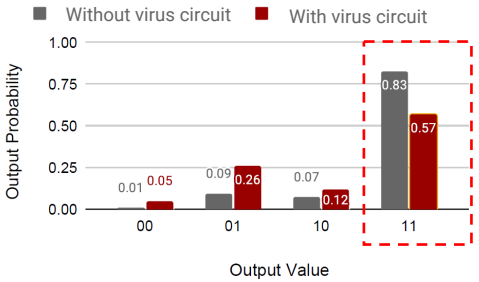


Reference:
Deshpande, et al. (HOST 2022)

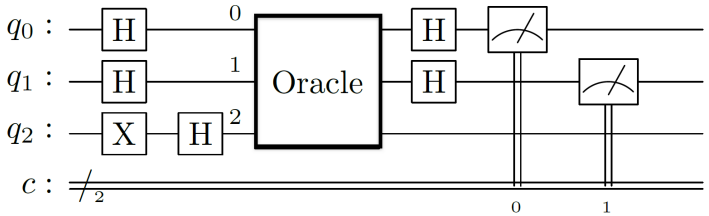
Changing Output Probability of Victim Circuits



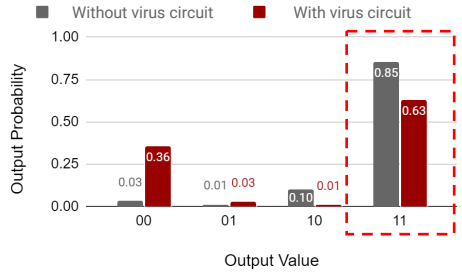
Grover's Circuit



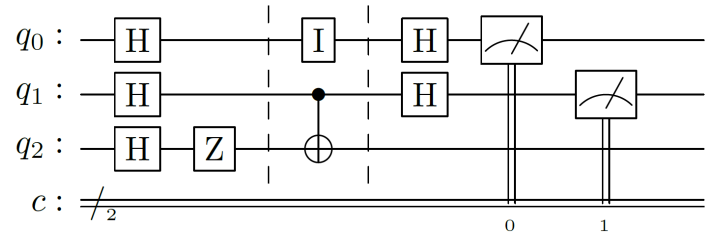
- Grover's Algorithm:
  - Provides Quadratic Speed up in Unstructured Search
  - The attack here is to lower the correct output probability and increase probability of random output
  - On a large scale, applications like traveling salesman problem, deadlock prevention could be attacked



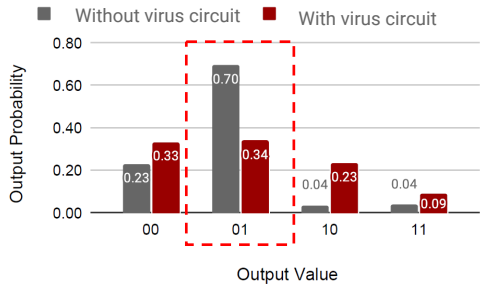
Deutsch-Josza Circuit



- Deutsch-Jozsa Algorithm:
  - Determines whether a given Oracle is constant or balanced
  - The attack here is to identify balanced oracle as a constant oracle



Bernstien-Vazirani Circuit



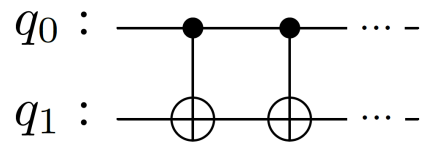
- Bernstein-Vazirani Algorithm:
  - Finds out a hidden string with a single query to Oracle
  - The attack here is retrieving incorrect secret string at the output

Reference:
Deshpande, et al. (HOST 2022)

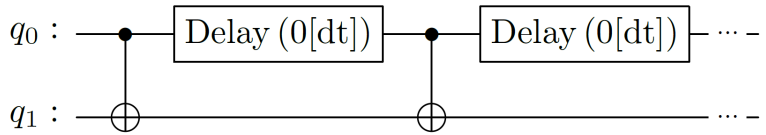
Candidate “Virus” Circuits

- Different possible circuits could be designed to generate excessive crosstalk, e.g.:

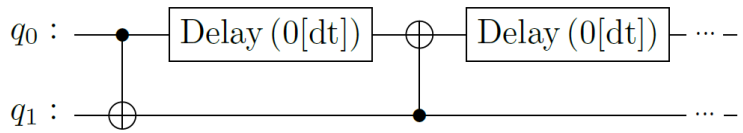
M1



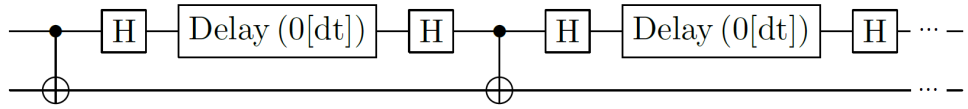
M2



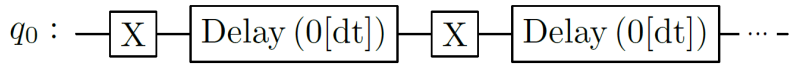
M3



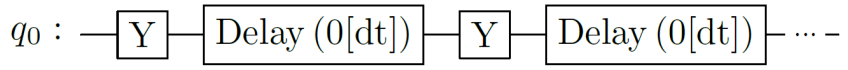
M4



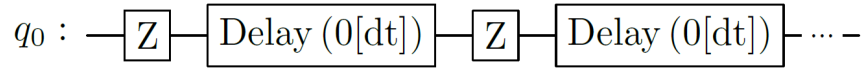
M5



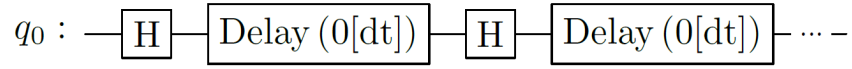
M6



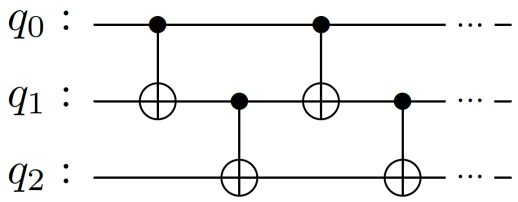
M7



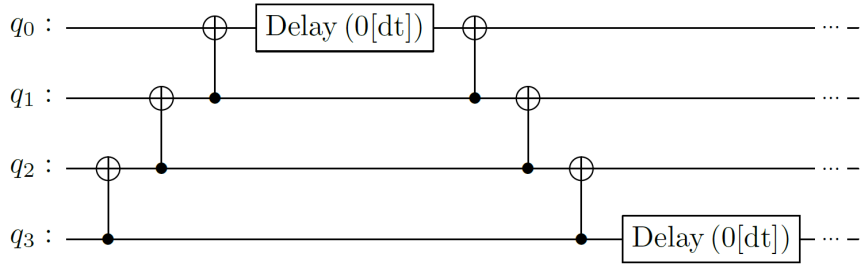
M8



M9



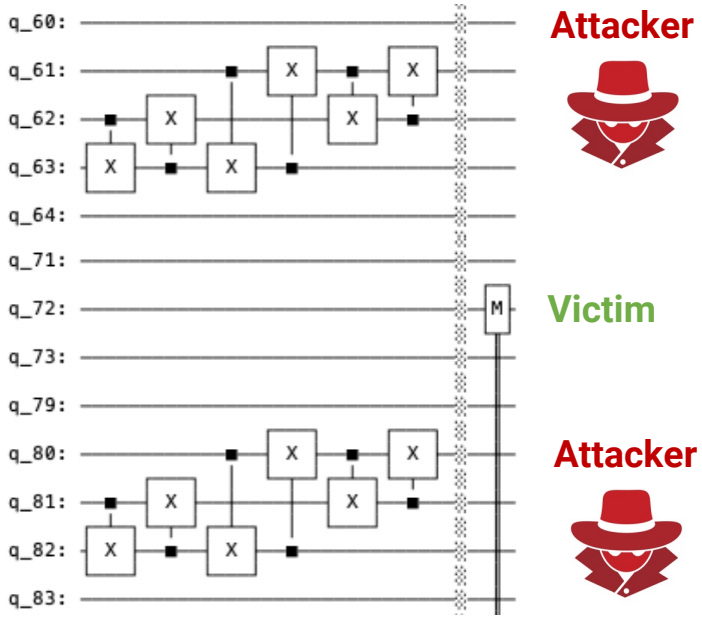
M10

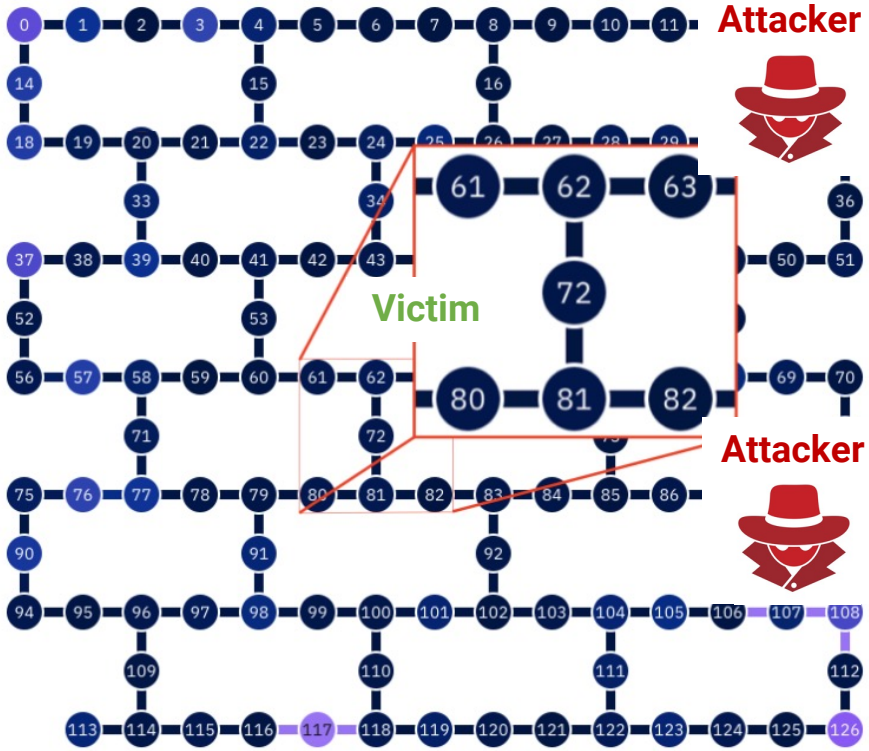


Reference:
Deshpande, et al. (HOST 2022)

Double-Sided Crosstalk Attacks

- Extension of the crosstalk attack is to perform the attack by leveraging qubits on two sides of the victim qubit:

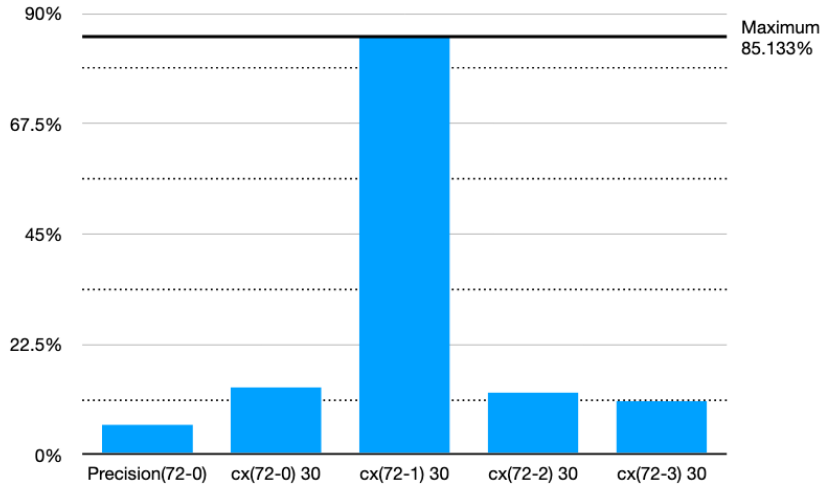




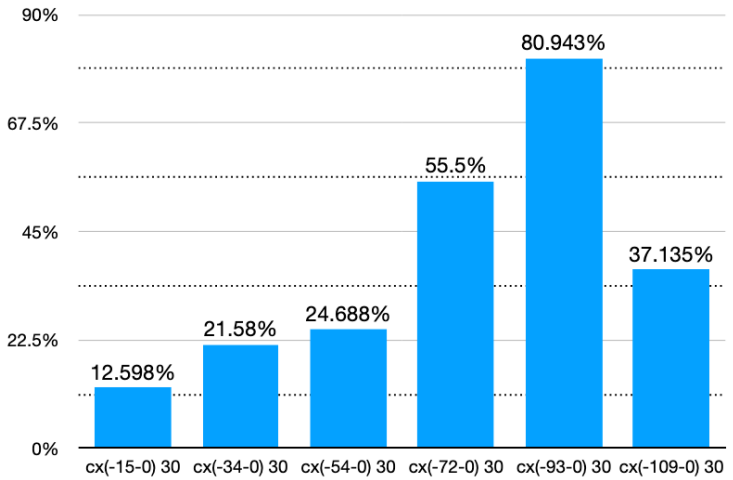
Reference:
Almaguer-Angeles, et al. (arXiv 2025)

Double-Sided Crosstalk Attack Success Rates

- On IBM Sherbrooke, 30 $CNOT$ gates around victim qubit 72 can cause the victim to flip from $|0\rangle$ to $|1\rangle$:



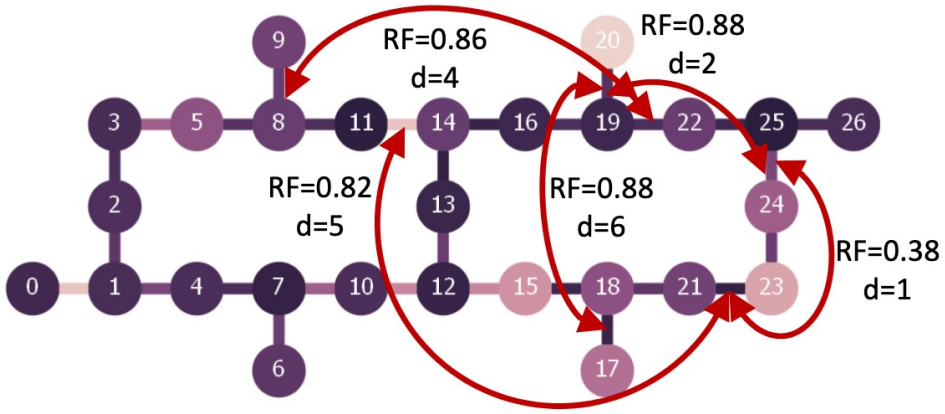
- As well as $|1\rangle$ to $|0\rangle$:



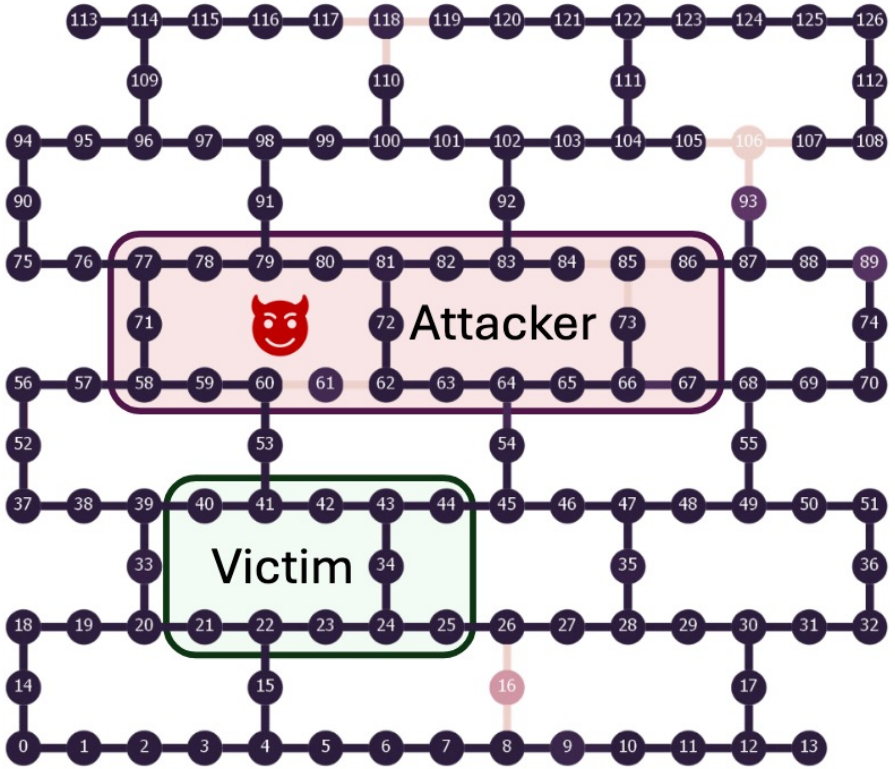
Reference:
Almaguer-Angeles, et al. (arXiv 2025)

High Crosstalk Coupling Pairs

- Beyond adjacent qubits and couplings, certain couplings display higher crosstalk than others, e.g., high crosstalk pairs on IBM Hanoi:



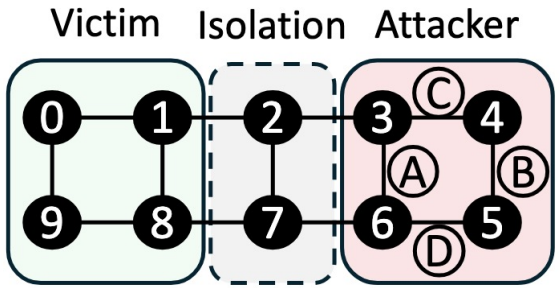
- Victim and attacker can 1 or 2 couplings away:

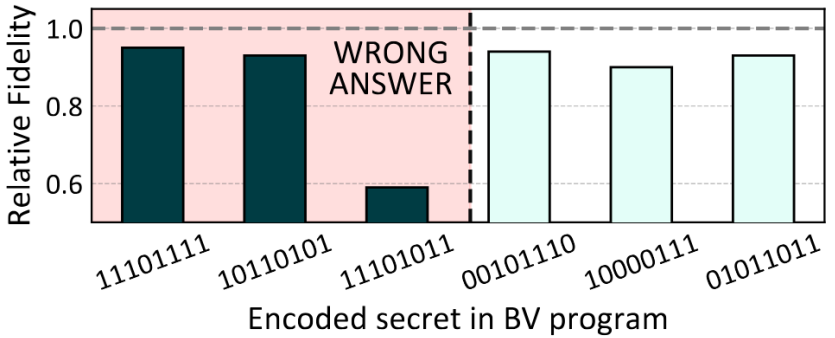


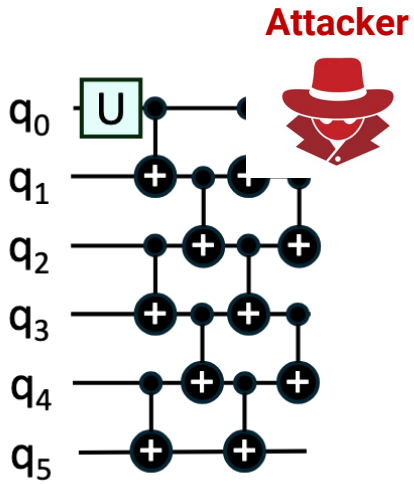
Reference:
Kumar, et al. (arXiv 2025)

Example Attacks on BV Circuits

- Attacker leverages repeated $CNOT$ gates to induce noise, even if there is isolation:



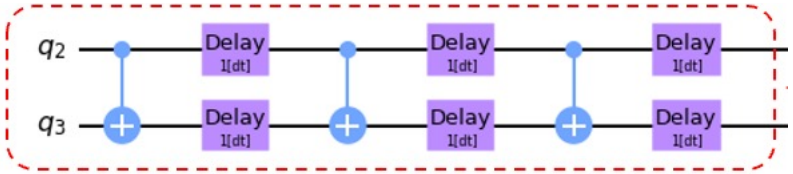




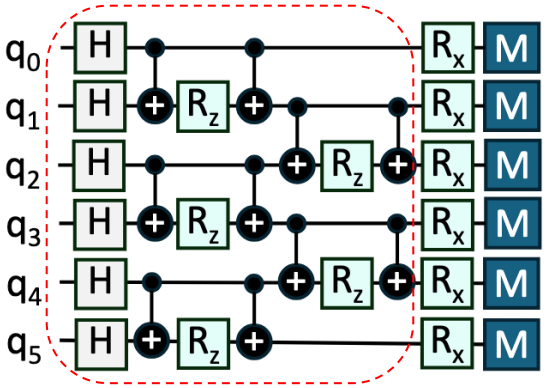
Reference:
Kumar, et al. (arXiv 2025)

Evolution of Obfuscation of Attacker Circuits

- Hiding the attacker circuit can help make the attacks more stealthy:



Insert delays

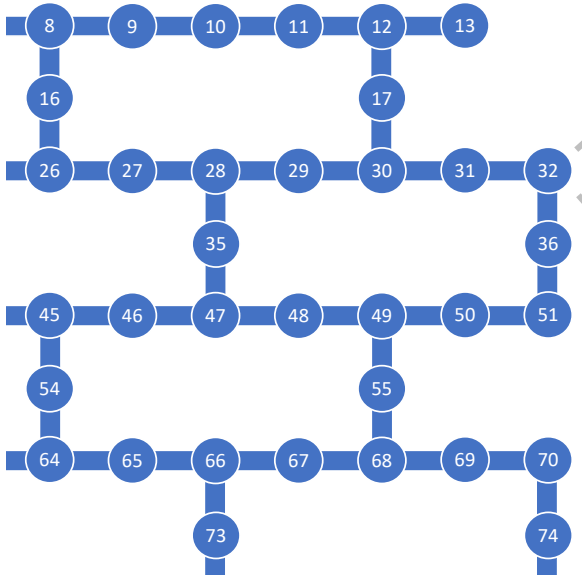


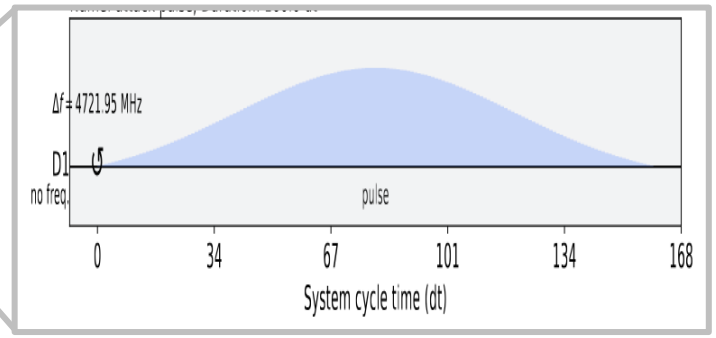
Make circuit look like QAOA circuit

Reference:
Deshpande, et al. (HOST 2022)
Kumar, et al. (arXiv 2025)

Novel Crosstalk Attacks with Custom Control Pulses

- In superconducting qubit quantum computers, each qubit has it's own frequency and other parameters:





Qubit control pulse for X gate

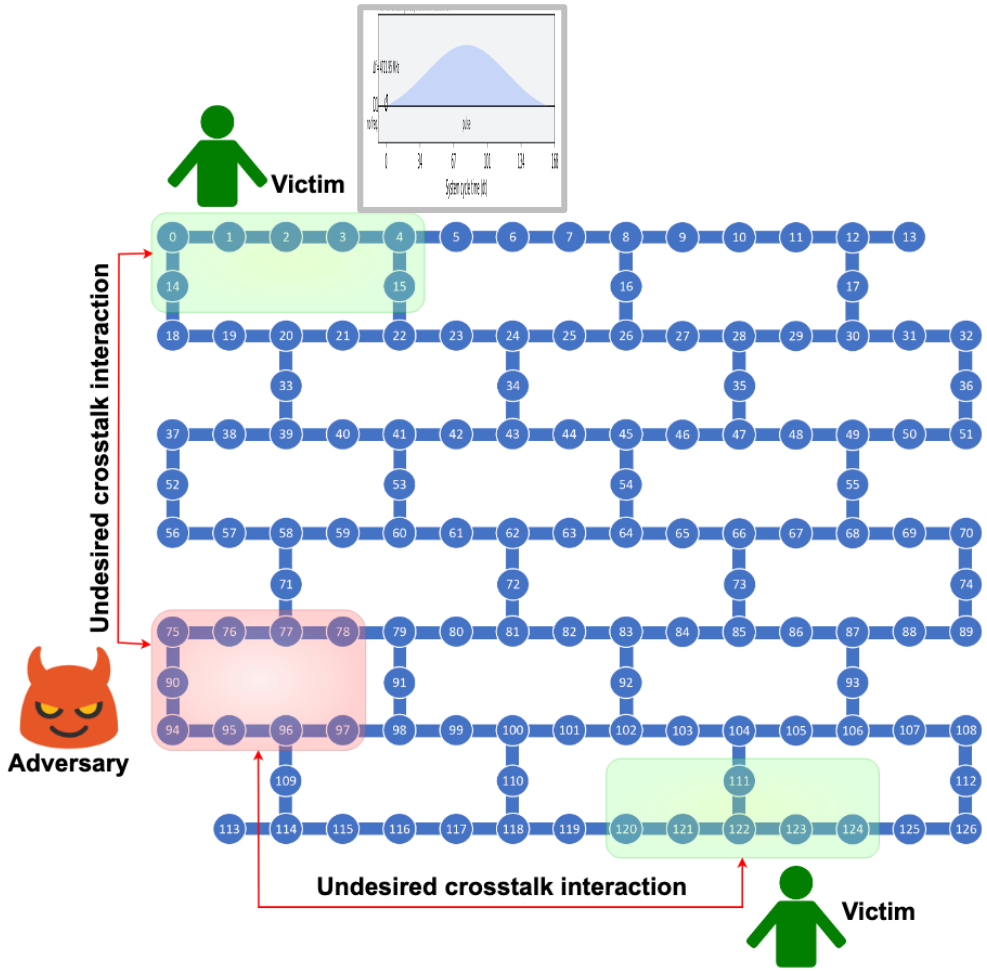
- Various qubit properties:

Qubit	Energy relaxation time T1 (μs)	Dephasing time T2 (μs)	Frequency (GHz)	Readout error rate
Q0	98.33551265	46.82808058	4.641200919	5.50000e-2
Q1	66.80352570	84.08716319	4.719992563	6.05e-2
Q2	77.96055113	88.27547132	4.761962111	4.04999e-2
Q3	93.81392649	86.04122557	4.687003753	5.04999e-2
Q4	57.44885729	40.31605568	4.923978085	4.00000e-2

Reference:
Tan, et al. (arXiv 2025)
IBM

Configuring Custom Control Pulse

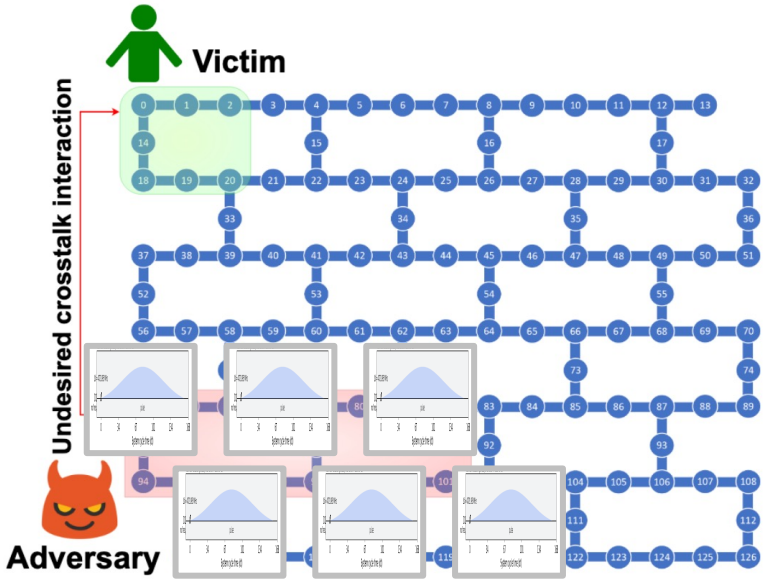
- Qiskit Pulse is one library that allows for configuration of custom pulses, usually to explore novel algorithm design
- **But can configure wrong pulse on purpose to explore crosstalk effects**
- *E.g. executing pulse for qubit 0 on qubit 75 has no effect on qubit 75, but may affect qubit 0 through crosstalk*



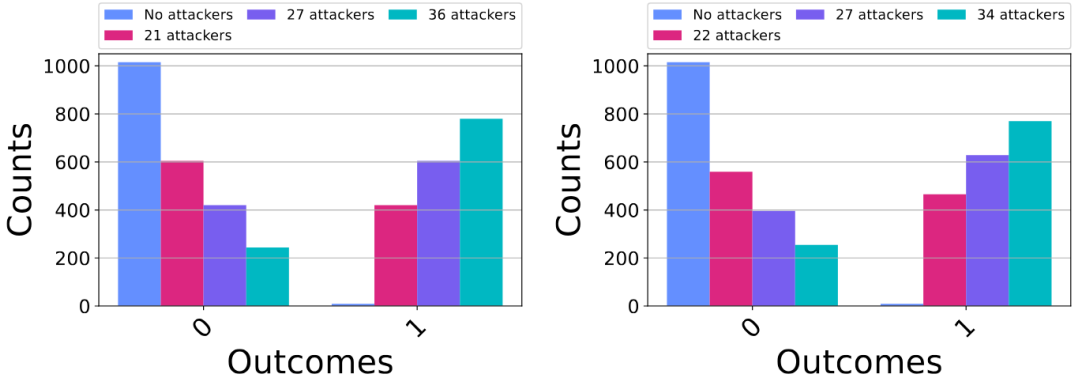
Reference:
Tan, et al. (arXiv 2025)

Long-Distance Crosstalk Attacks

- Repeating the ”wrong” control pulses on many distant qubits can lead to disruption of the victim qubit and circuits running on victims qubits:



- Attack results at distance 13 and 17:



Reference:
Tan, et al. (arXiv 2025)

Crosstalk Attacks Summary and Outlook

Many attacks have been demonstrated:

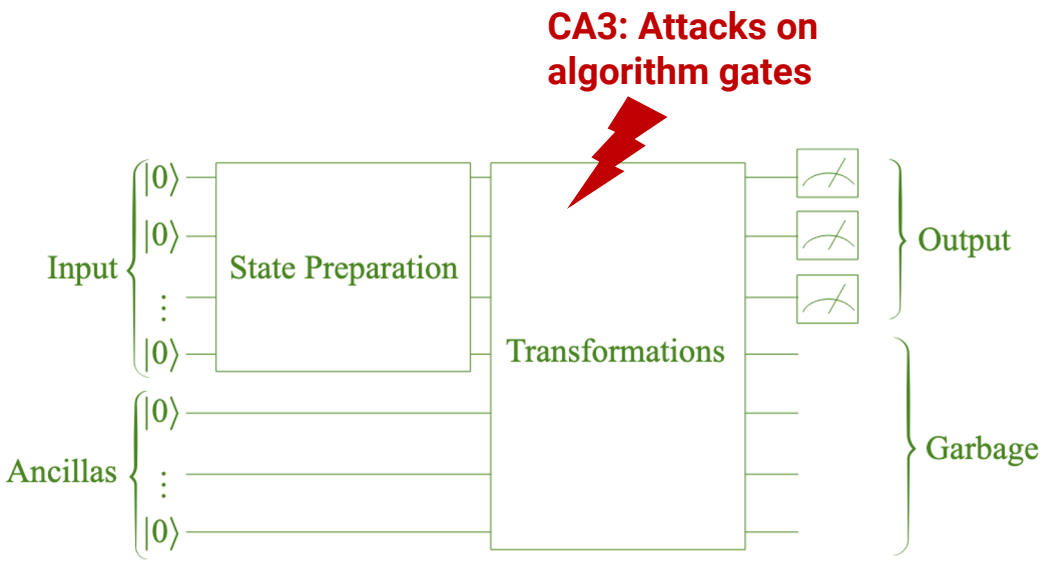
- Single-side crosstalk (since about 2021)
- Double-side crosstalk (2025)
- Crosstalk with custom pules (since about 2024)

Various defenses have been proposed in parallel:

- Quantum computer antivirus (since about 2021)
- Others, e.g., isolation or scheduling changes

Future directions in this research:

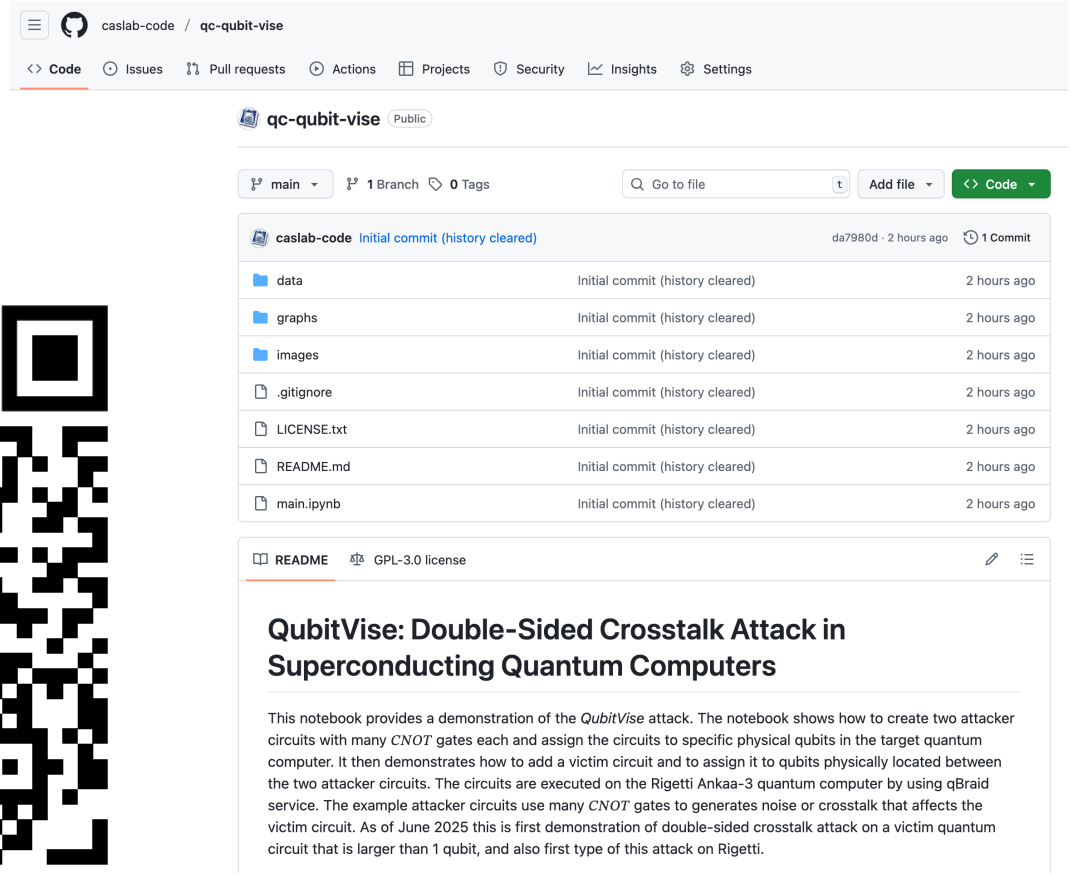
- Longer-distance crosstalk
- Cross chiplet crosstalk attacks
- Integration with multitenancy



No quantum computer provider publicly uses multi-tenant configuration as of 2025

Demo of Crosstalk Effect

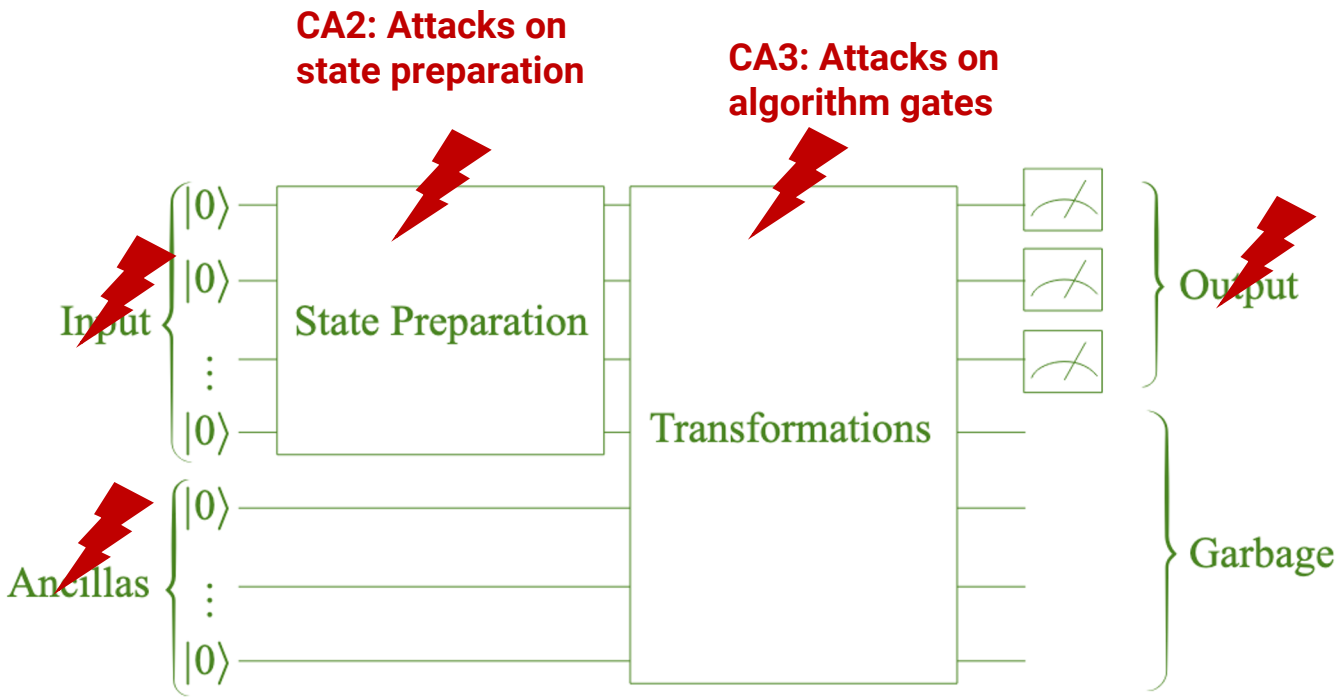
- **QubitVise: Double-Sided Crosstalk Attack in Superconducting Quantum Computers**
  - https://github.com/caslab-code/qc-qubit-vise
  - Complete code for the evaluation of double-sided crosstalk
  - (requires qBraid account, could be adapted to use Amazon Braket directly without qBraid)



Fault Injection on Reset Gates

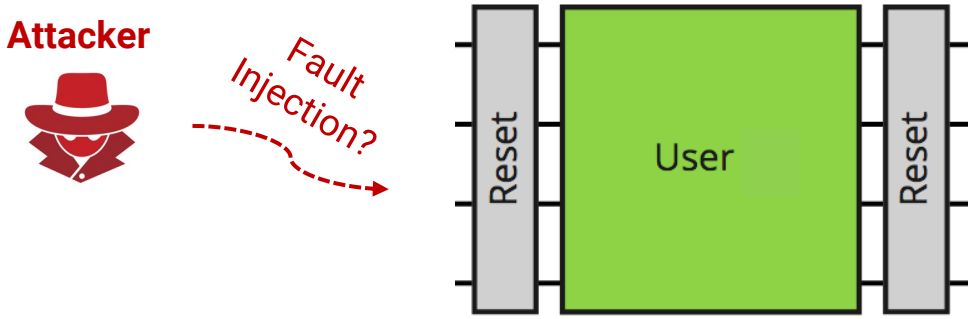
CA1.A: Attacks on initial input qubits states

CA1.B: Attacks on initial ancilla qubits states



Affecting Initial Qubit State When Reset Gates are Imperfect

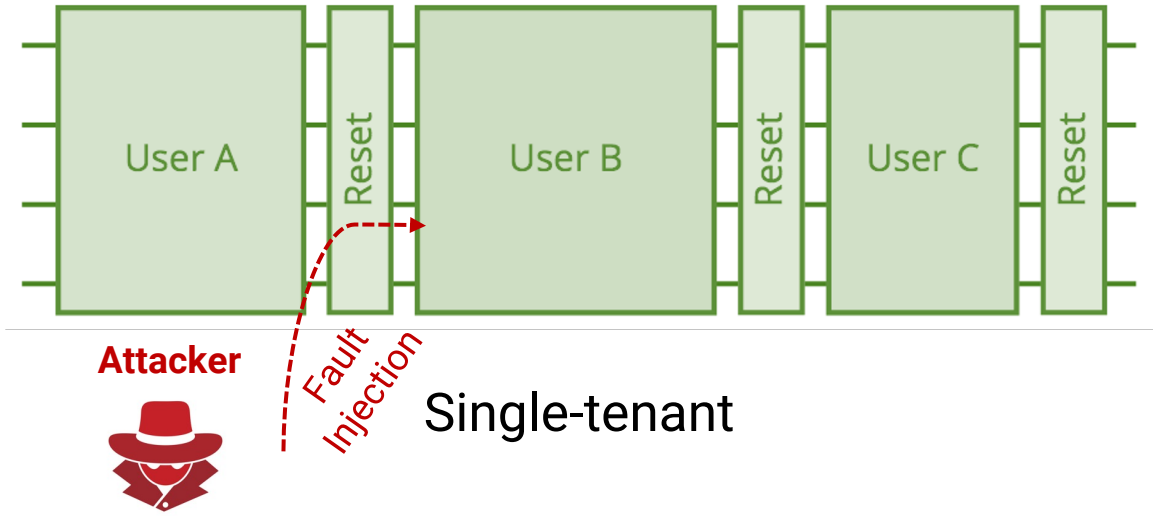
- Incorrect operation of the reset gate can create information leaks for user programs
  - If attacker runs before victim, could cause incorrect initial state, possibly affecting the computation
  - If attacker runs after victim, could recover some information about the final qubit state



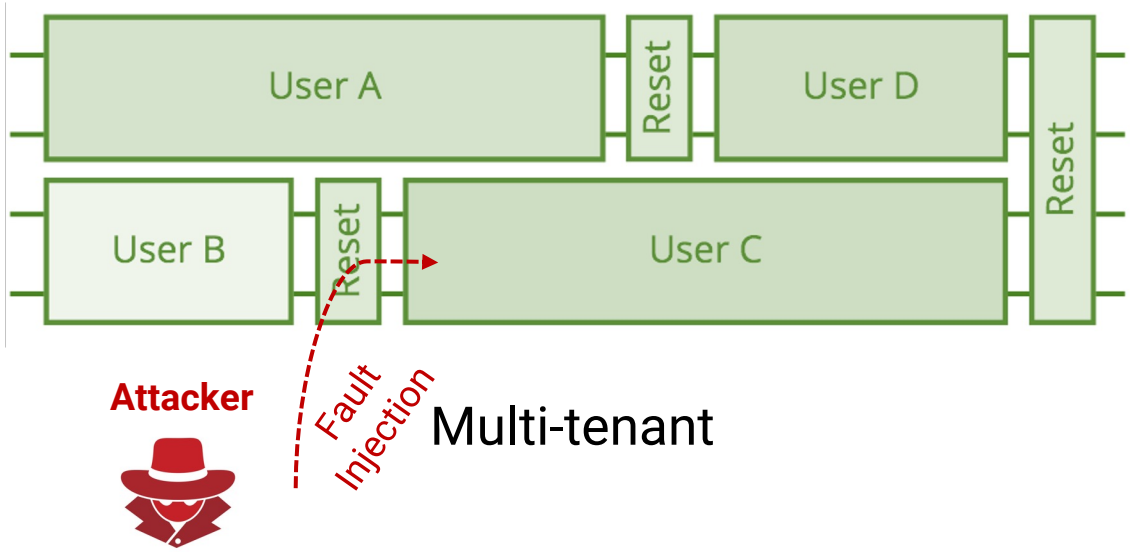
Reference:
Xu, et al. (CCS 2023)

Affecting Initial Qubit State When Reset Gates are Imperfect

- It is possible to share the quantum computing hardware in single- or multi-tenant ways:



Single-tenant



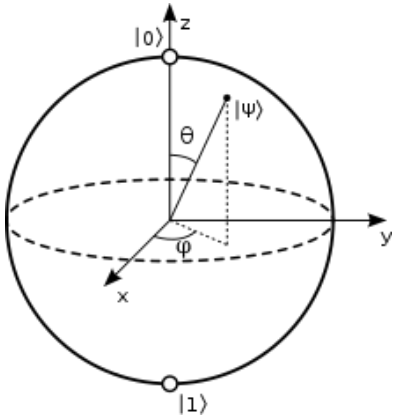
Multi-tenant

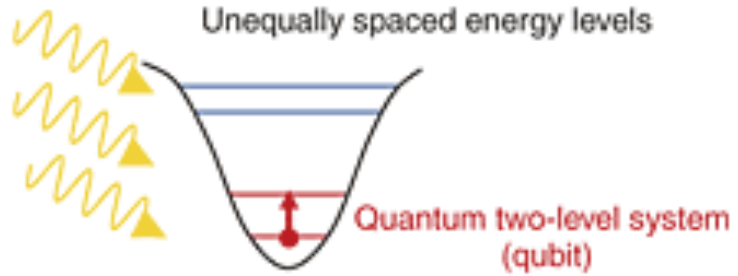
Reference:
Xu, et al. (CCS 2023)

Revisiting Qubit Definition

Qubit: basic element of quantum computation

- Classical bits: 0 and 1
- Qubits: superposition of 0 and 1
- Abstraction of a two-state quantum-mechanical system:
  - $|0\rangle$ Ground state
  - $|1\rangle$ First excited state



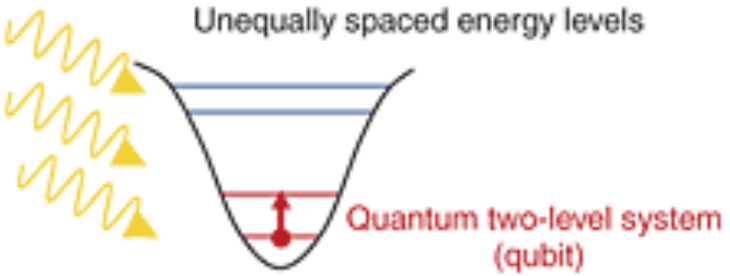


However, the physical realizations of qubits allow for states with higher energies

Reference:
Xu, et al. (CCS 2023)

The π Gate

- The π gate is used to excite the state from the lower to higher energy state
- Gate $\pi_{g,e}$ exchanges the amplitudes on two energy levels:
  - If $|\psi\rangle = a_g|g\rangle + a_e|e\rangle$ then $\pi_{g,e}|\psi\rangle = a_e|g\rangle + a_g|e\rangle$
- Note, the X gate is also the $\pi_{0,1}$ gate that can excite $|0\rangle$ to $|1\rangle$, and also can change the state from $|1\rangle$ to $|0\rangle$

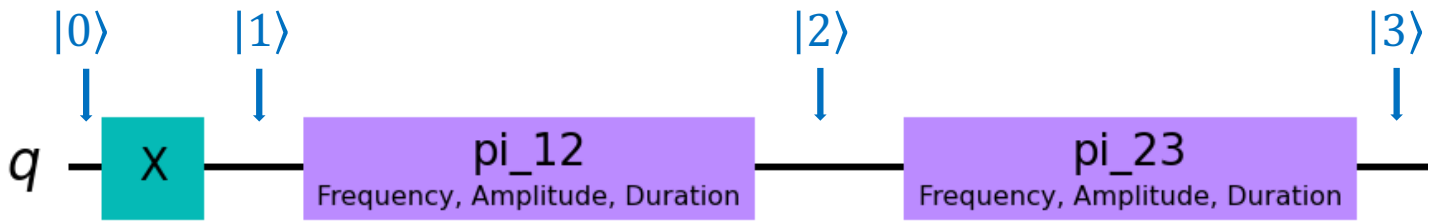


Reference:
Xu, et al. (CCS 2023)

Setting Higher Energy States

- IBM Quantum, custom pulse gates are executable, and thus higher energy states may be obtained
  1. Define $\pi_{1,2}$ and $\pi_{2,3}$ gates
  2. Start with `X` gate (i.e. $\pi_{0,1}$ gate)
  3. Apply $\pi_{1,2}$ and $\pi_{2,3}$ gates sequentially to raise the energy level

- Gate for going from `|0>` to `|3>` energy level:

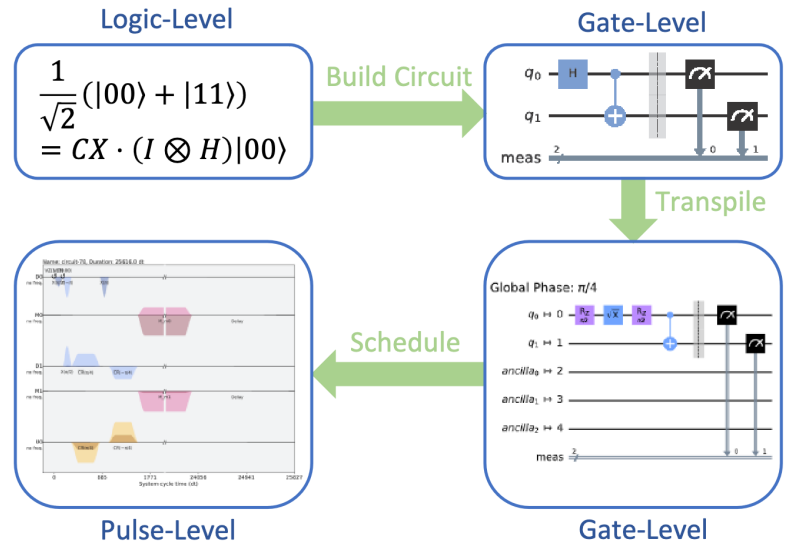


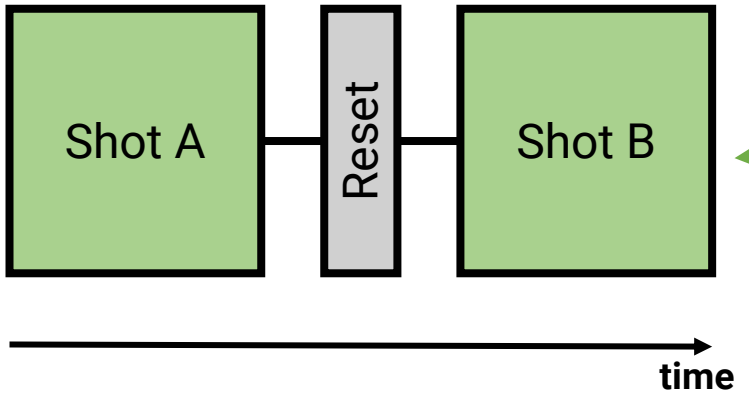
Reference:
Xu, et al. (CCS 2023)

Review of Quantum Computer Workflow

- User's code is transpiled into gate-level code and eventually scheduled as control pulses sent to the quantum computer

- The pulse-level circuits are scheduled on the quantum computer
  - Each circuit execution is called a 'shot'

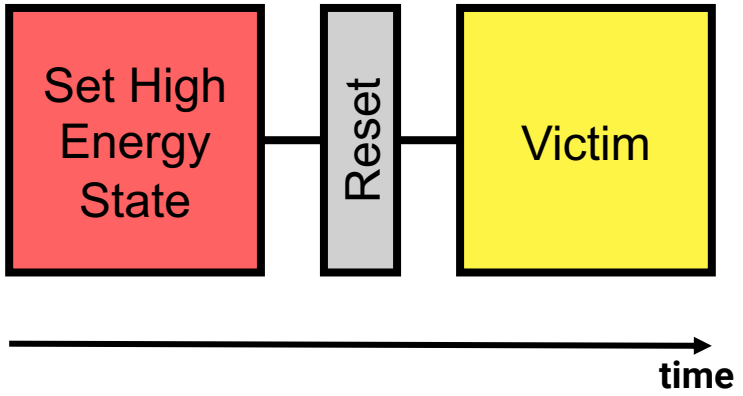




Reference:
Xu, et al. (CCS 2023)

Threat Model

- Attacker scheduled before victim, shot by shot, on the same qubit
- Set qubit state to high energy
- Let victim execute after reset

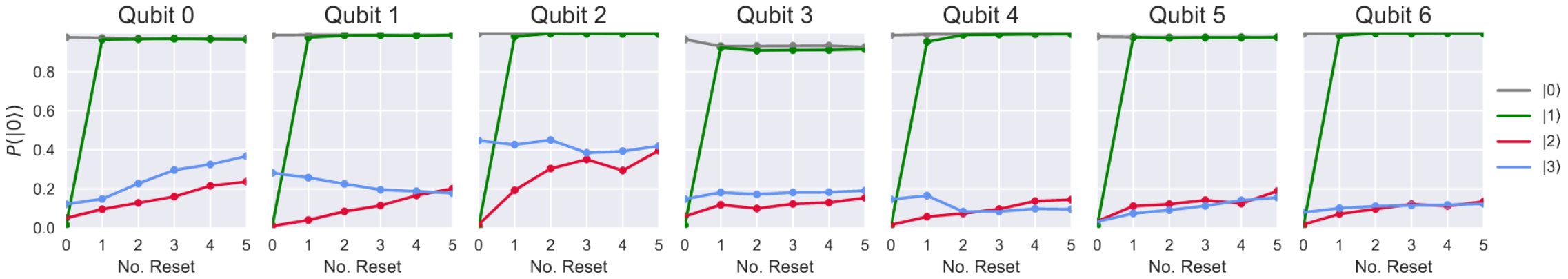


Reference:
Xu, et al. (CCS 2023)

Impact of Higher Energy States

- Basic gates are only calibrated considering two-level system, they do not have the expected effect on higher energy states

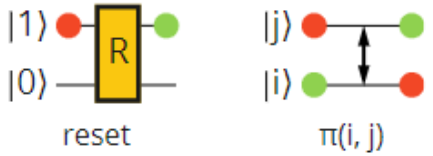
- **Example of effect on reset gate**
  - Probability of measuring |0> after different number of reset gates
  - Reset is not effective against |2> and |3> states



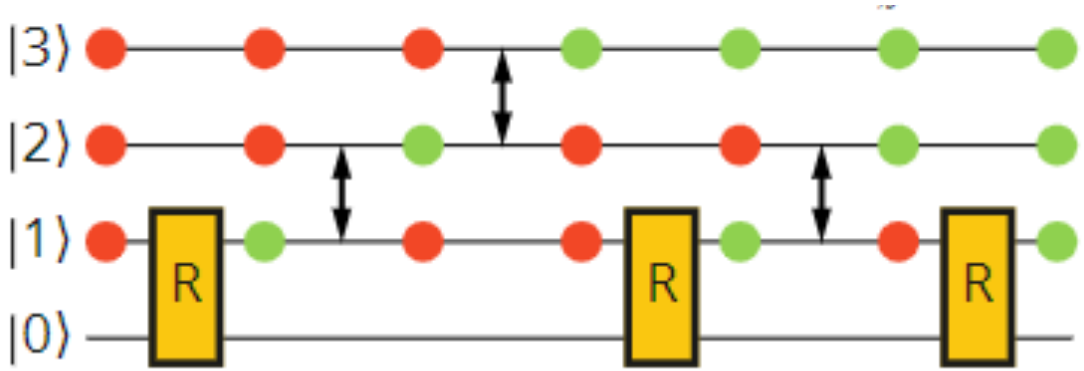
Reference:
Xu, et al. (CCS 2023)

Design of Cascading Secure Reset (CSR) Gate

- Basic idea: sequentially bring down high energy states to lower energy states, down to $|1\rangle$ which can be eliminated using the native reset gate
- Building blocks: reset and Π gates



- Cascading Secure Reset (CSR):

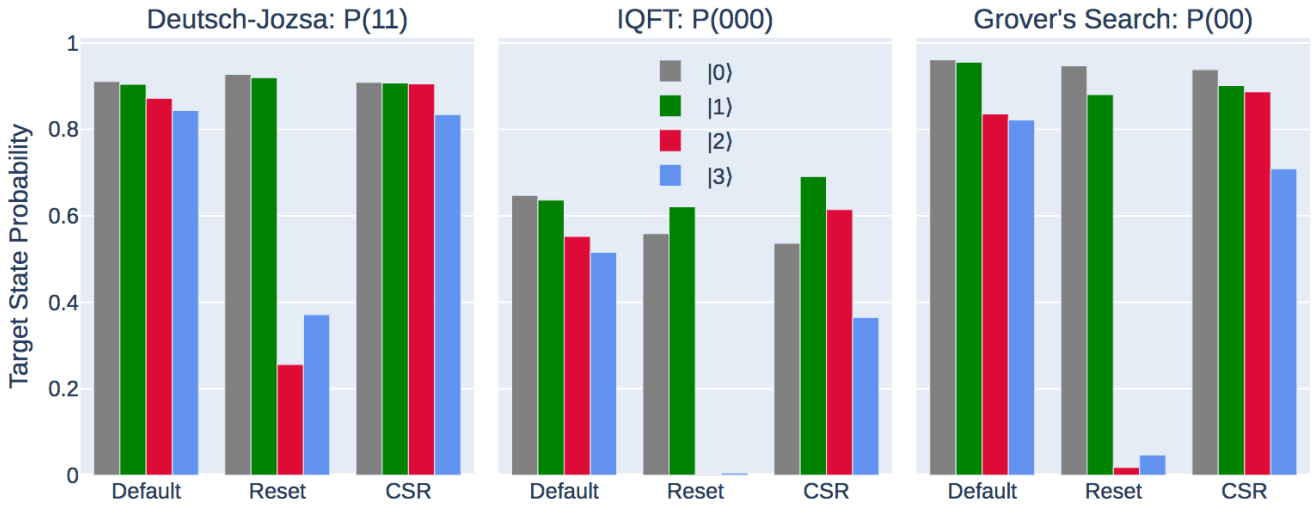


Reference:
Xu, et al. (CCS 2023)

Impact of Higher Energy States

- Higher energy states affect all gates, and circuits made of these gates

- **Example of effect on results of different algorithms**
  - Deutsch-Jozsa
  - Inverse quantum Fourier transformation
  - Grover's search



Reference:
Xu, et al. (CCS 2023)

Fault Injection Attacks on Reset Gates Summary and Outlook

Many attacks have been demonstrated:

- Reset gate attack
  (since about 2022)

Various defenses have been proposed in parallel:

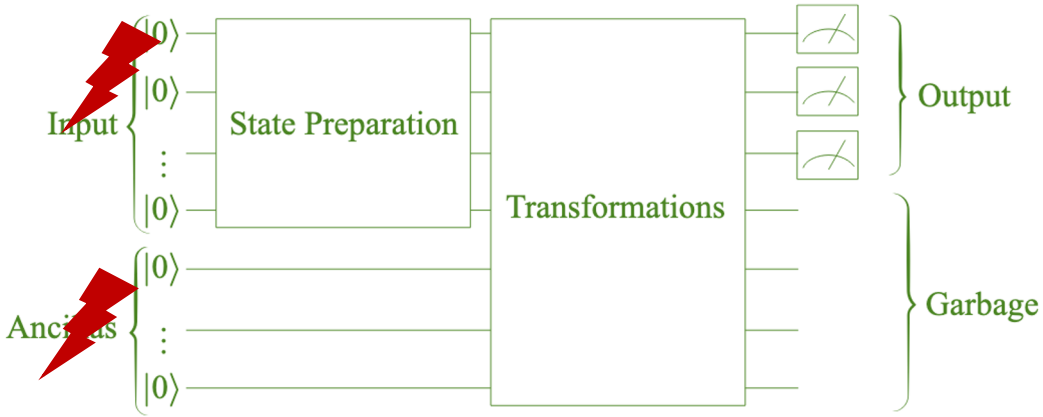
- Cascading reset gate
  (since about 2022)

Future directions in this research:

- Attacks with even higher energy states
- Reset mechanisms in other superconducting architectures

CA1.A: Attacks on initial input qubits states

CA1.B: Attacks on initial ancilla qubits states



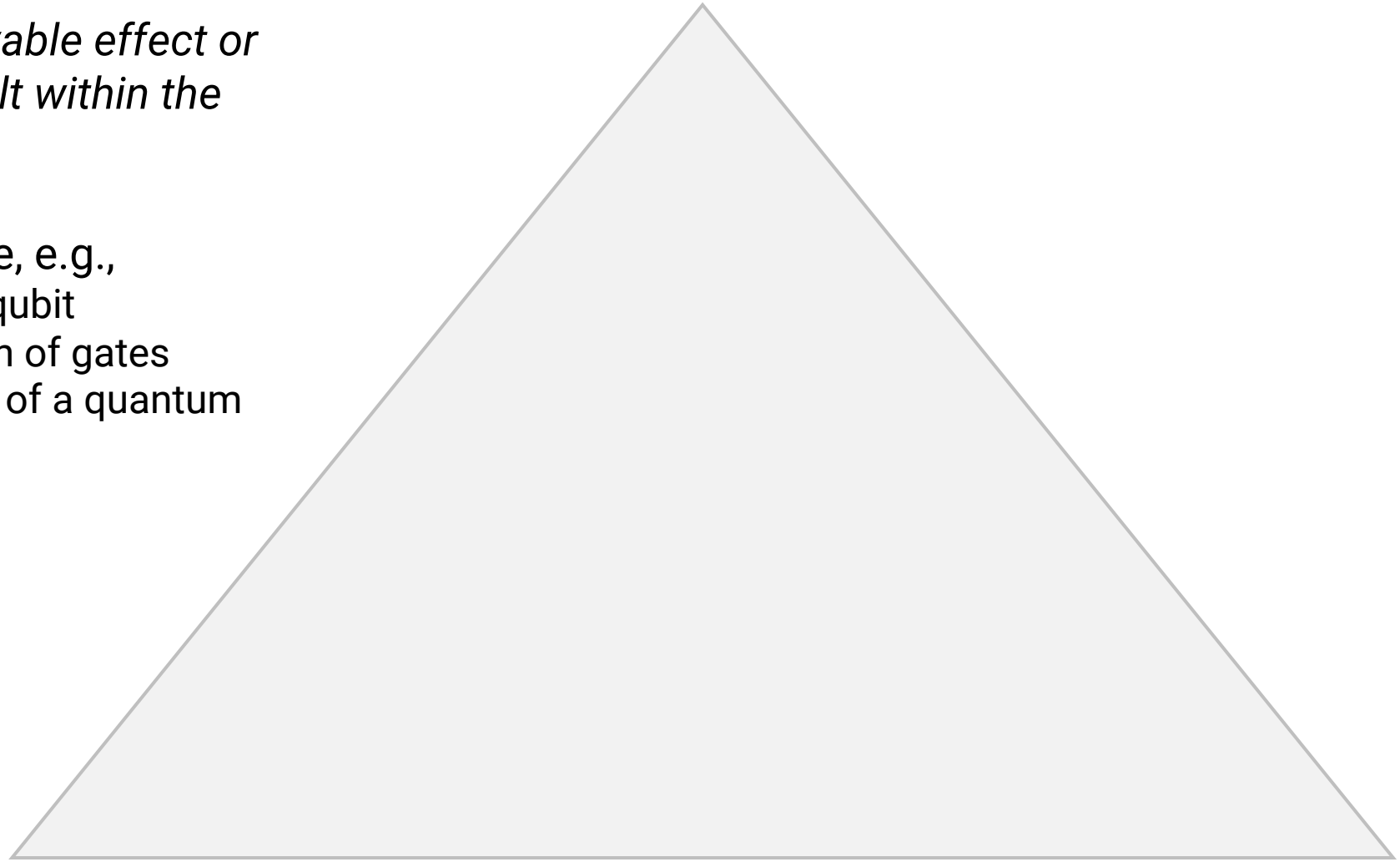
Fault Injection Attack Classification

- A classification can help enumerate different types of fault injection attacks
- Classification has many parts:
  - **Fault manifestation** – denotes the observable effect of a fault injeciton
  - **Fault target** – denotes the equipment or location where the fault is injected
  - **Fault model** – specifies the model or type of fault and how the fault behaves
  - **Fault bound** – defines the number of faults considered
  - **Fault lifespan** – specifies duration for which a fault persists in a system

Fault Manifestation

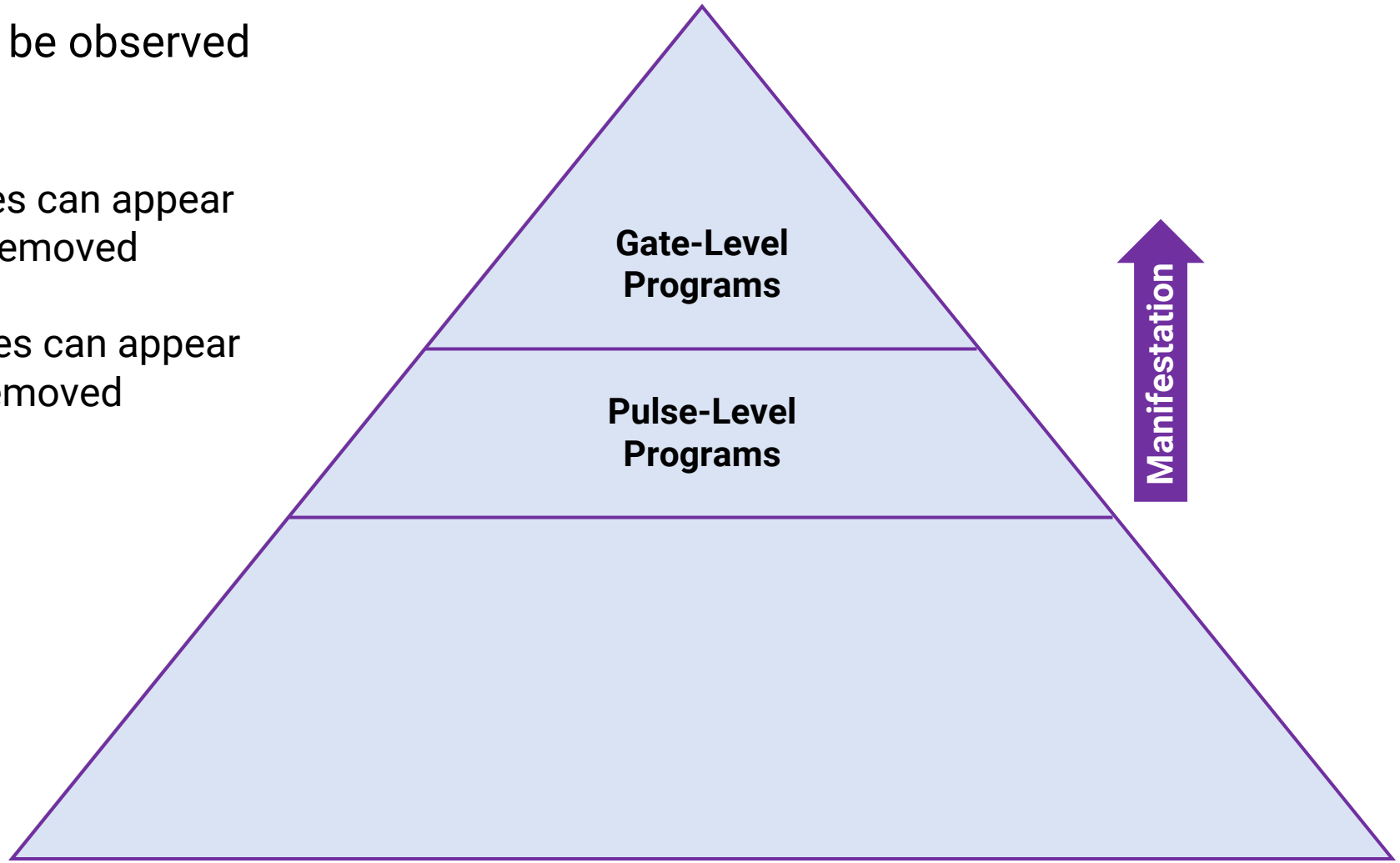
Fault manifestation: *the observable effect or consequence of an injected fault within the quantum system*

- Fault manifestations include, e.g.,
  - Changes in the state of a qubit
  - Alterations in the operation of gates
  - Deviations in the outcome of a quantum algorithm



Fault Manifestation

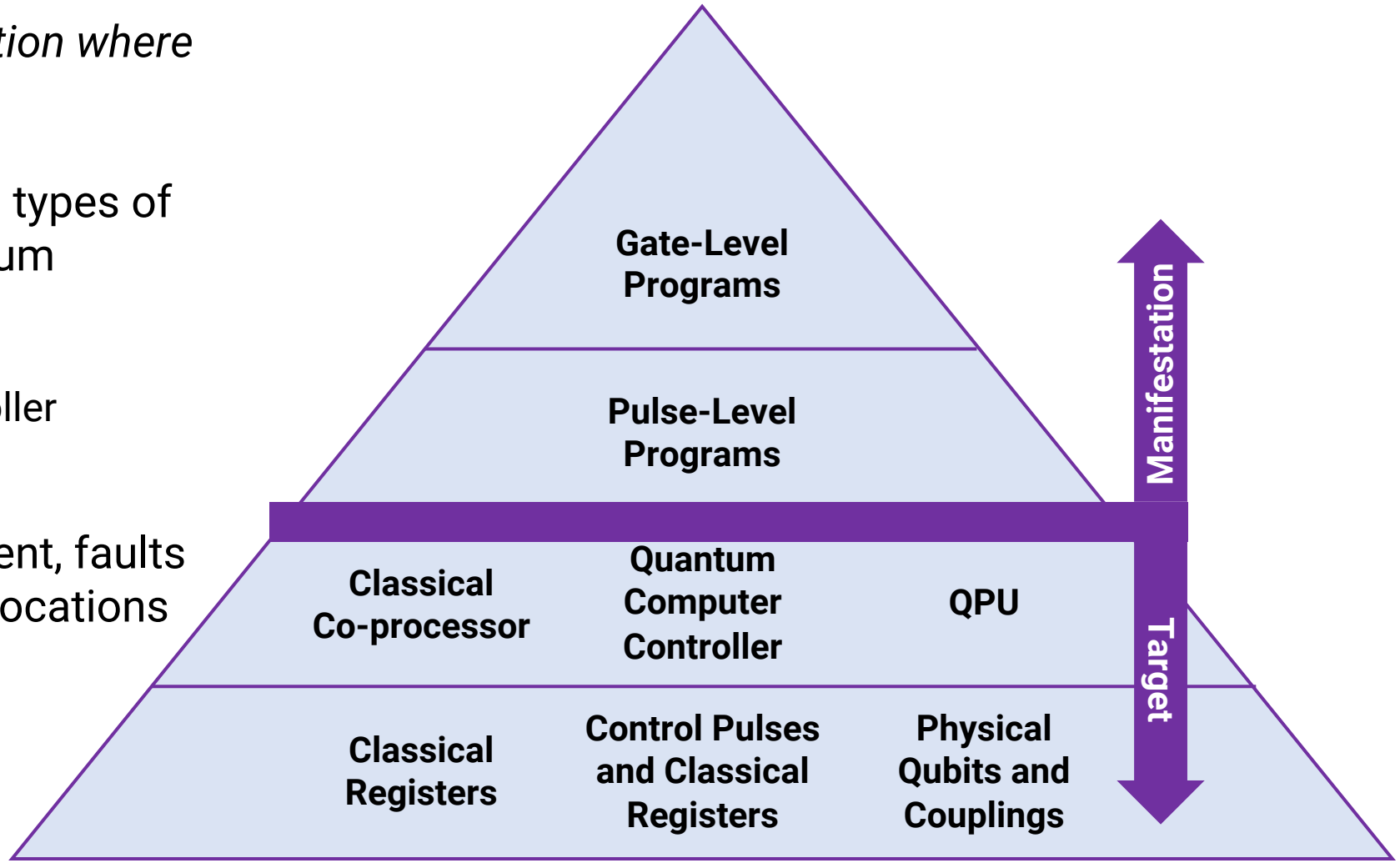
- The fault manifestation can be observed at different levels:
  - **Gate-level program** – gates can appear to be modified, added, or removed
  - **Pulse-level** – control pulses can appear to be modified added or removed



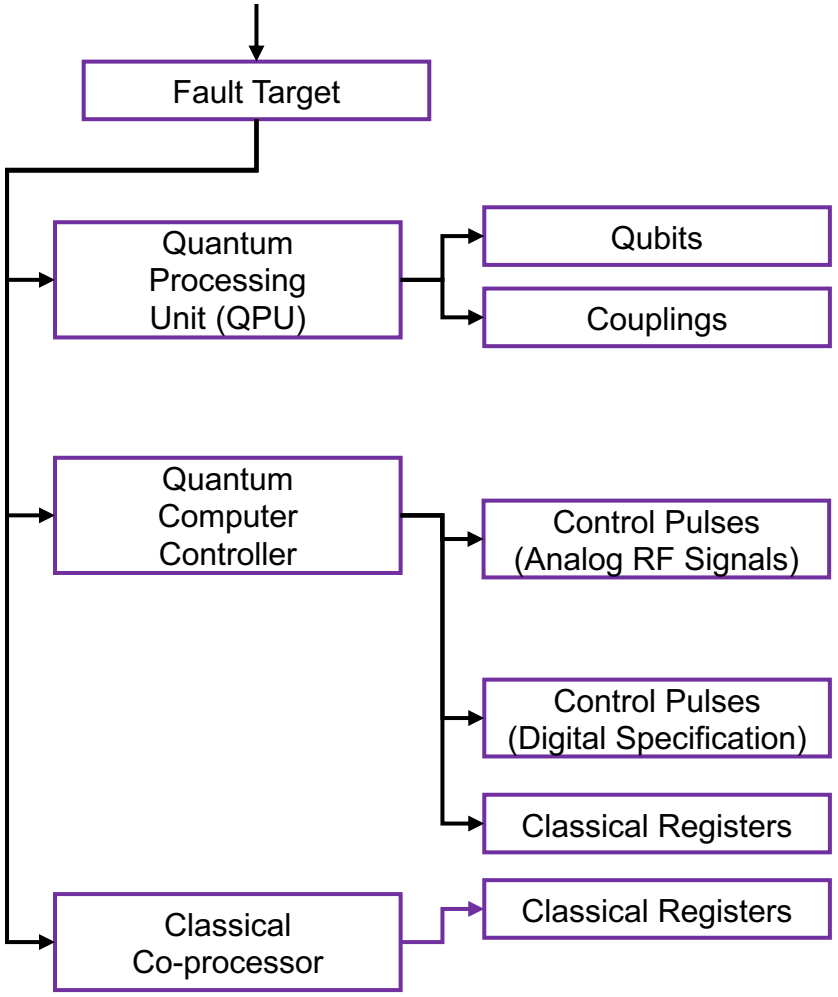
Fault Target

Fault target: *equipment or location where the fault is injected*

- Fault targets can be various types of equipment within the quantum computing system:
  - Classical co-processor
  - Quantum computer controller
  - QPU
- Within each type of equipment, faults can further target different locations within that equipment

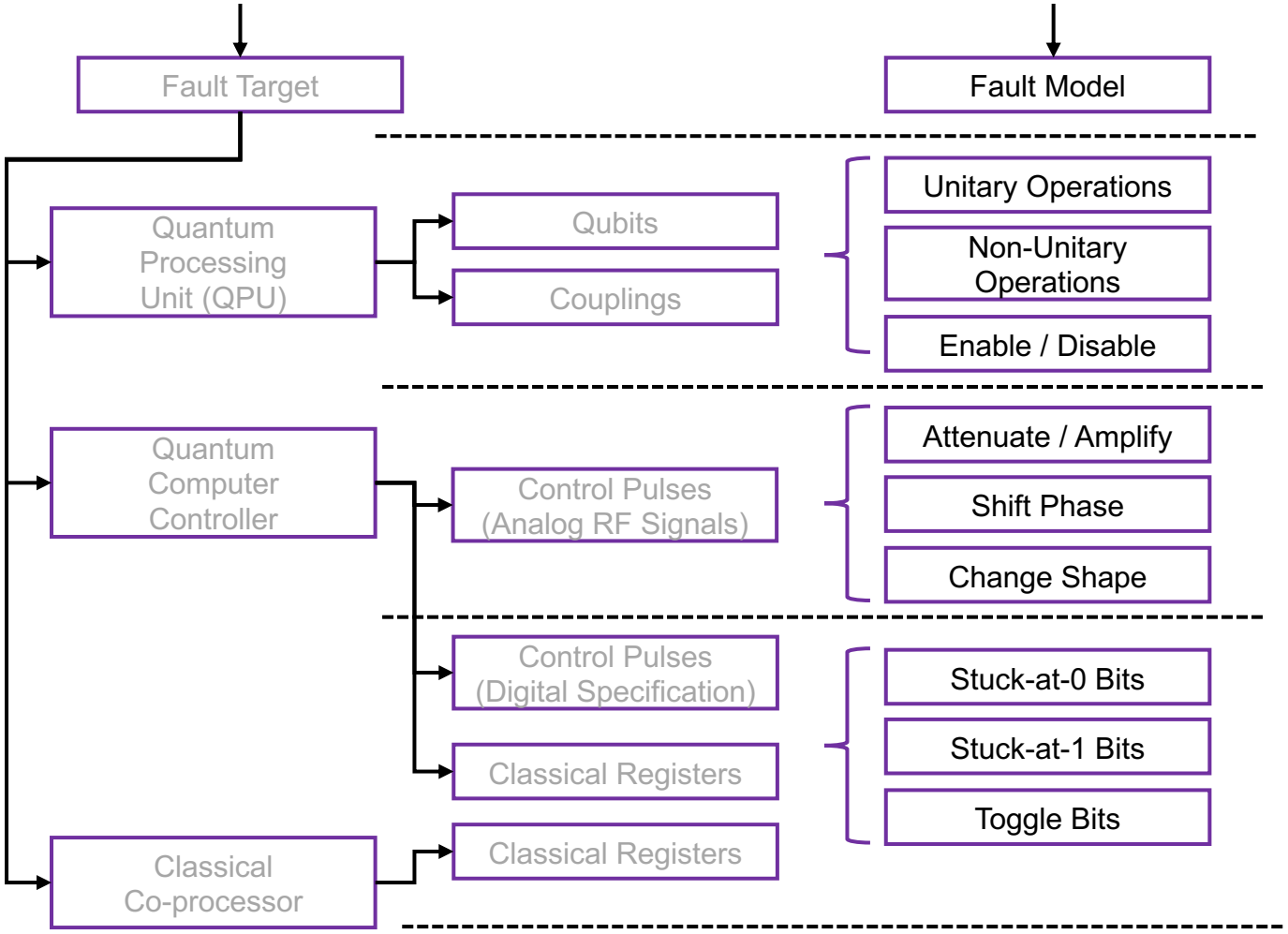


Fault Target Classification



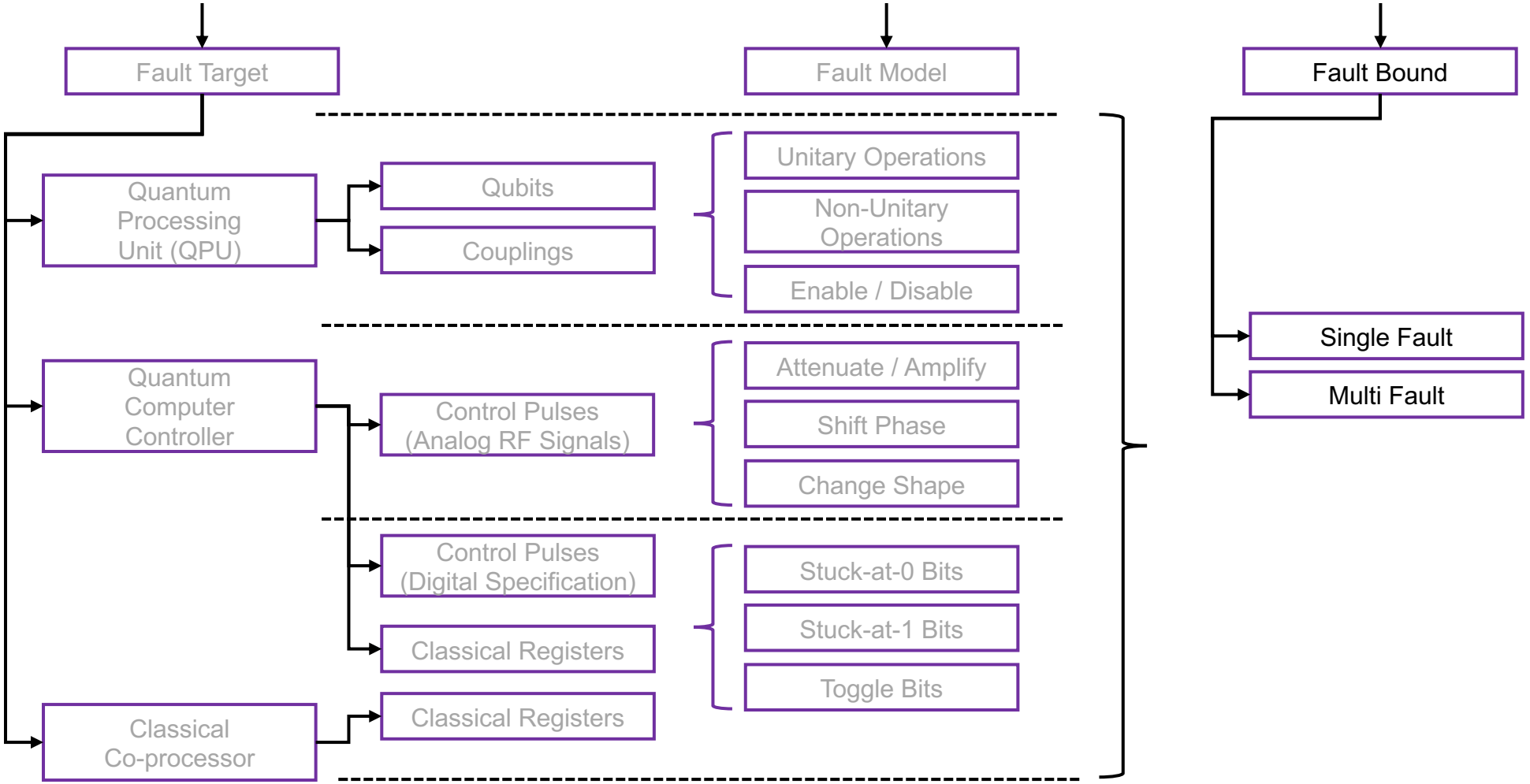
- Fault target is again the equipment or location where the fault is injected
- Within each equipment, there are many components and locations where faults can be injected
  - Different low-level sub-components or sub-locations can be targeted
  - Targets can be very implementation and equipment specific and differ for each system

Fault Model Classification

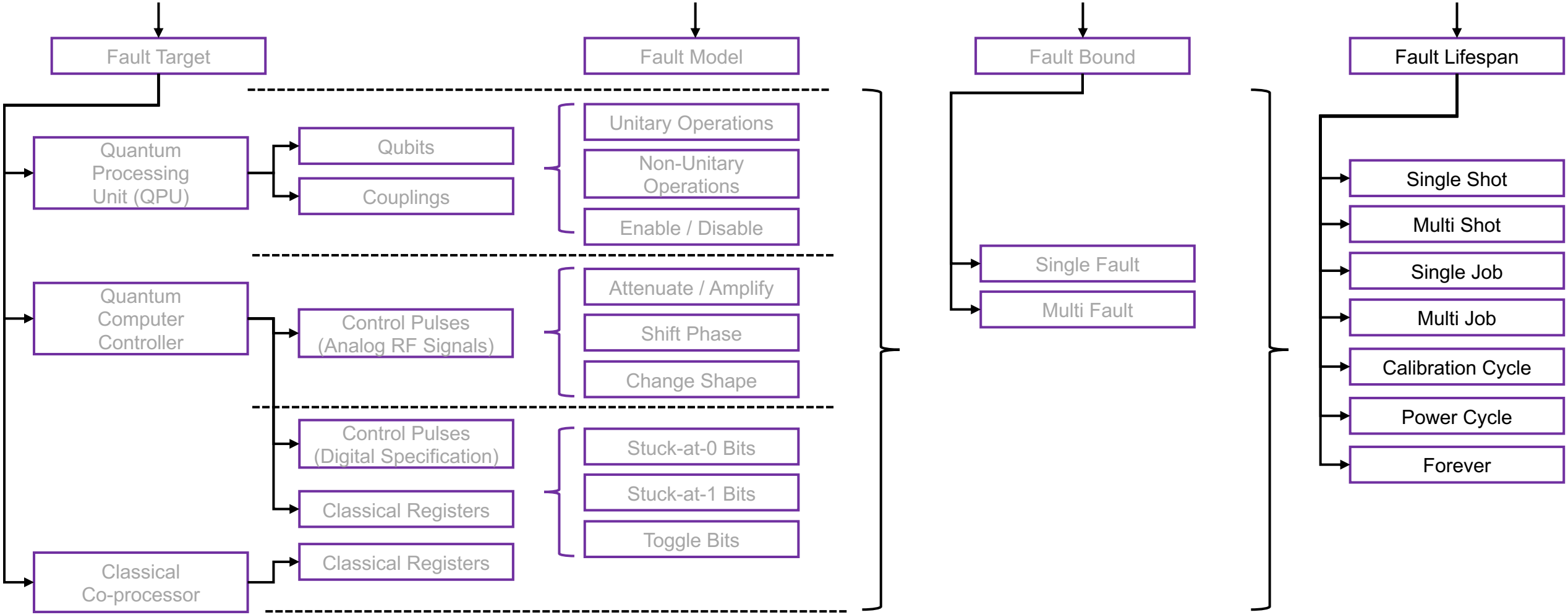


- Fault model is again the model or type of fault and how the fault behaves
- Fault models are specific to each target and equipment

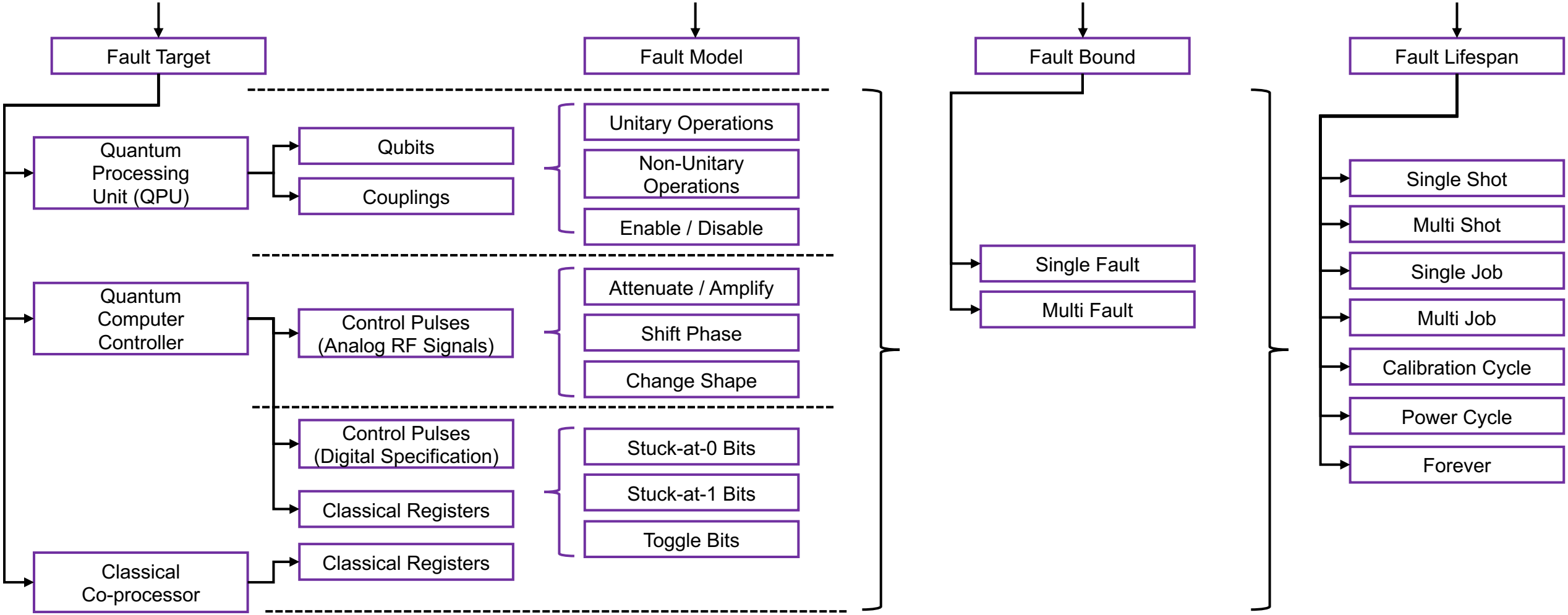
Fault Bound Classification



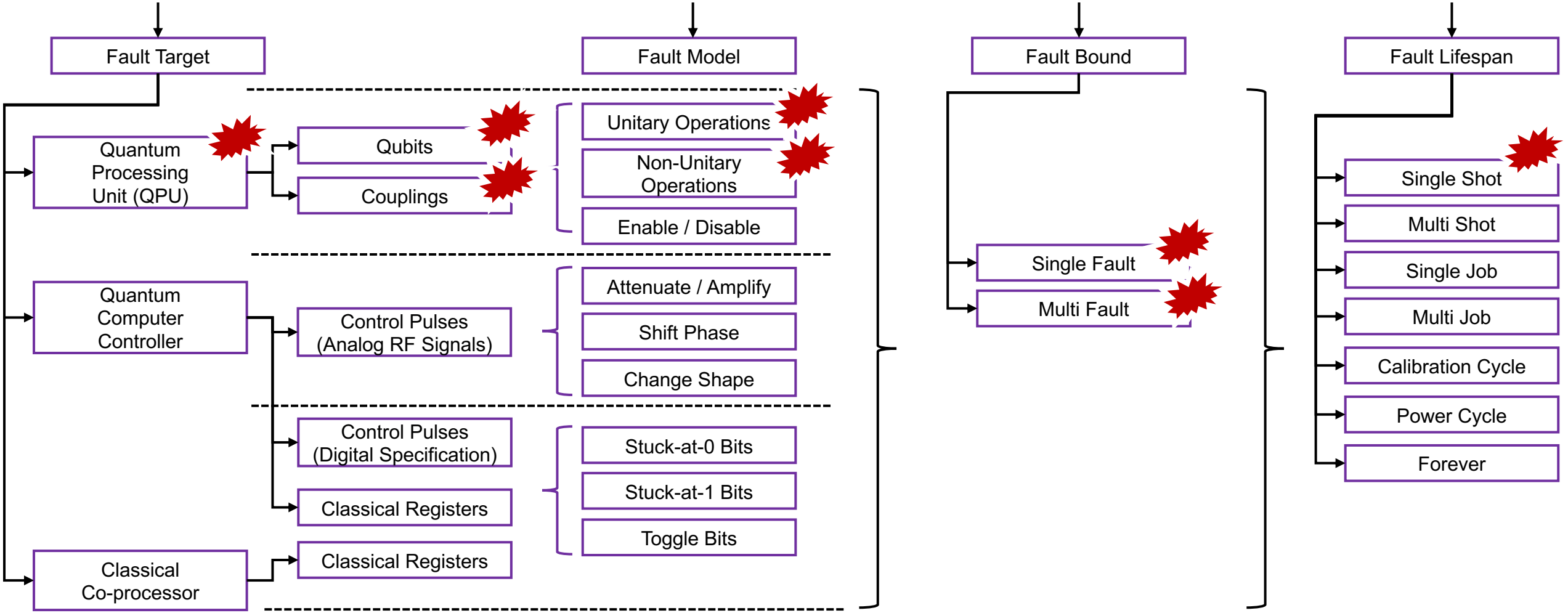
Fault Lifespan Classification



Fault Injection Attack Classification



Fault Injection Attacks that Have Been Explored



Tutorial Organization

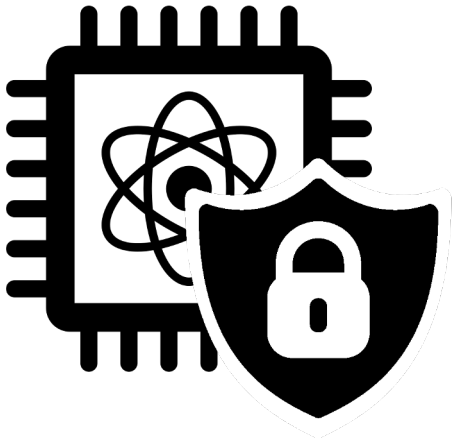
10:00am – 11:30am – Tutorial Part 1

- Motivation: Need for Securing Quantum Computers
- Brief Introduction to Classical Security Topics
- Overview of Threats to Quantum Computing Systems
- Fault Injection and Classification for NISQ Systems

11:30am – 1:00pm – Lunch

1:00pm – 2:30pm – Tutorial Part 2

- Side Channels in NISQ Systems
- Quantum Computer Infrastructure Security
- Quantum Computer Software Security
- Trusted Execution Environments for NISQ Systems
- Fault-Tolerant Quantum Computing (FTQC) Security



Tutorial on Security of Quantum Computing Systems

Side Channels in NISQ Systems

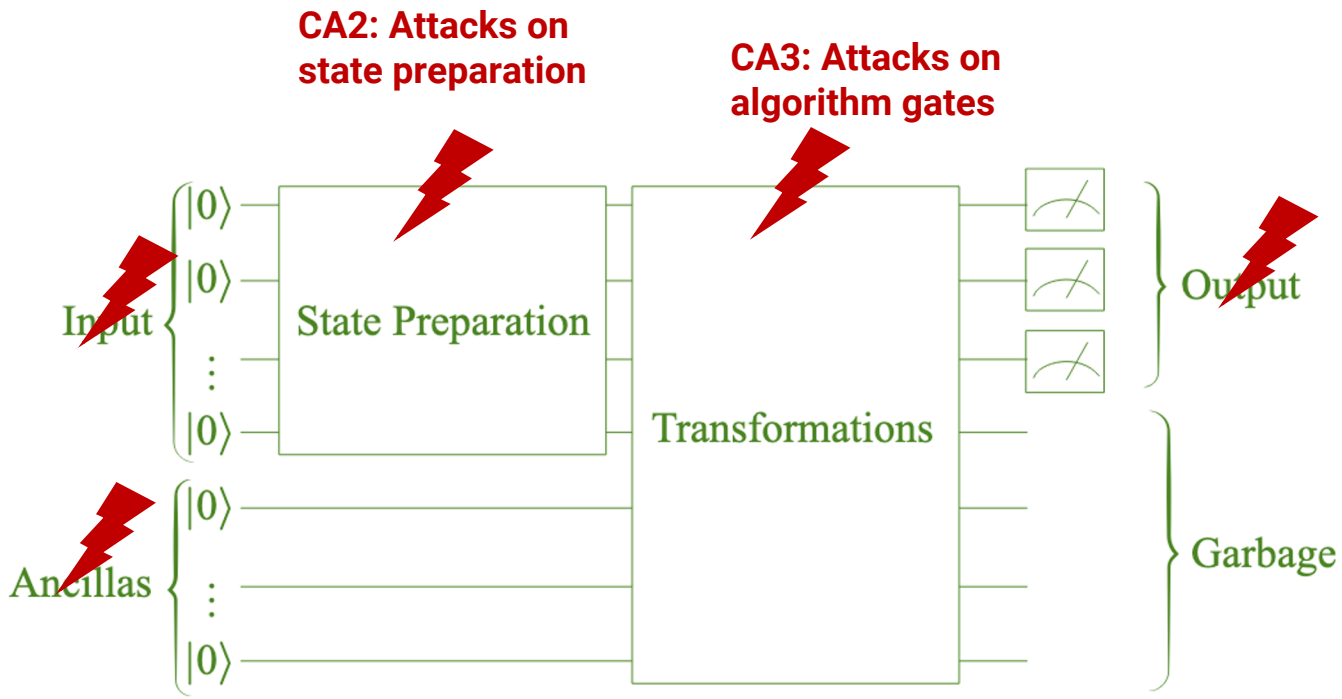
Computer Architecture and Security Lab (CASLAB)

Northwestern University

Side Channels Through Reset Gates

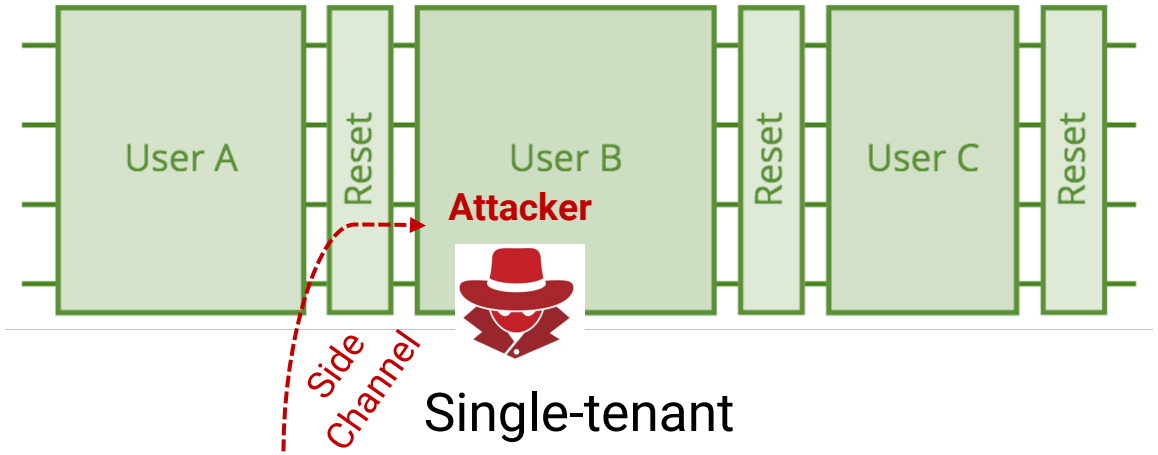
CA1.A: Attacks on initial input qubits states

CA1.B: Attacks on initial ancilla qubits states

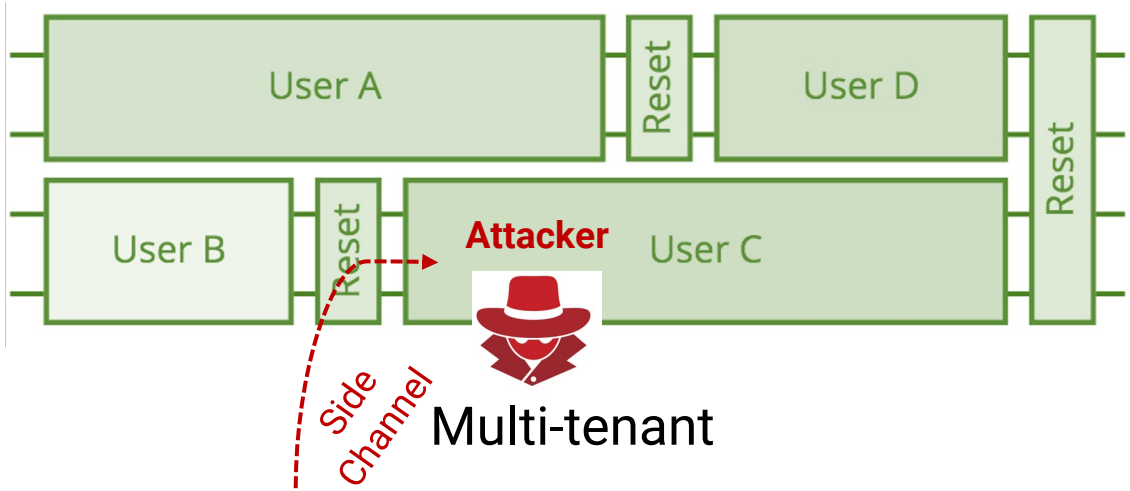


Affecting Qubit Outputs When Reset Gates are Imperfect

- It is possible to share the quantum computing hardware in single- or multi-tenant ways:



Single-tenant

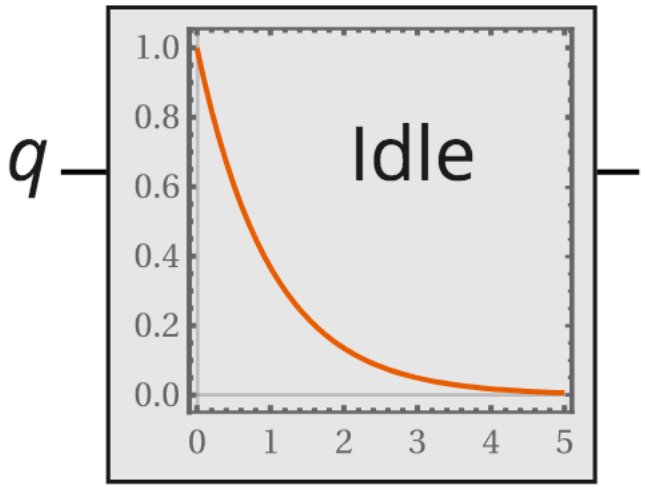


Multi-tenant

Example Means to Reset Qubits in Superconducting Computers

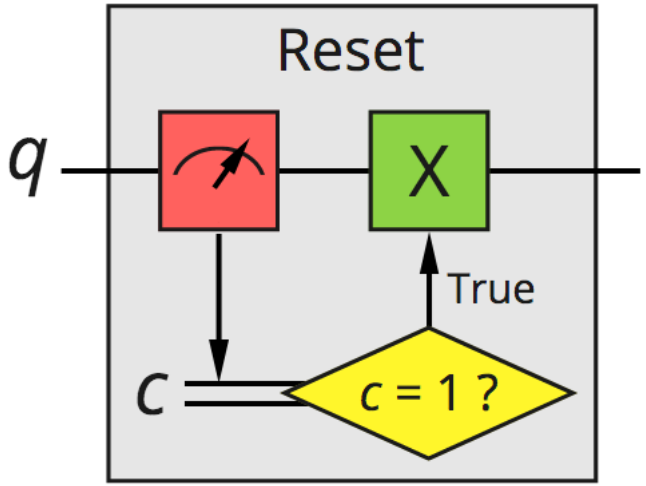
Thermalization (`250µs – 1ms`)

- Good: Low state retention
- Bad: Poor scaling with `T1`



Active reset (`1µs`)

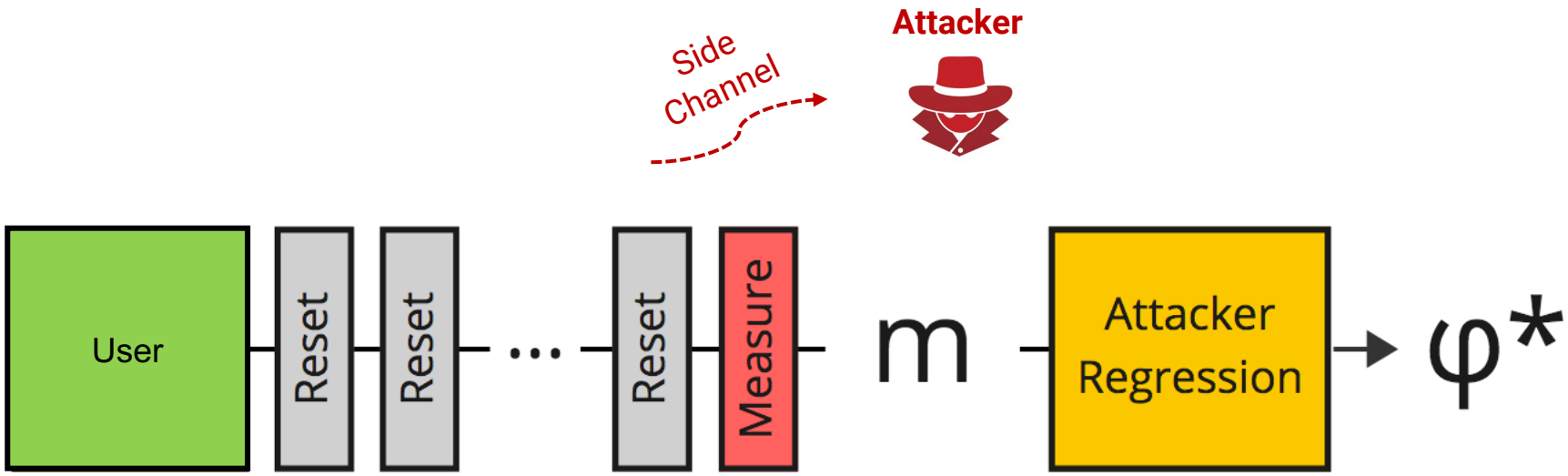
- Good: Fast and scales well
- Bad: Detectable state leakage



Reference:
Mi, et al. (CCS 2022)

Recovering Information Across Reset Operation

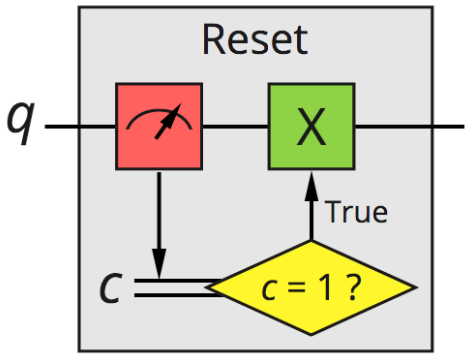
- Attacker scheduled after victim, shot by shot, on the same qubit
- Aim to correlate qubit state $|\Psi\rangle$ with post-reset measurement m
- Build model that infers $|\Psi\rangle$ given m and number of resets r

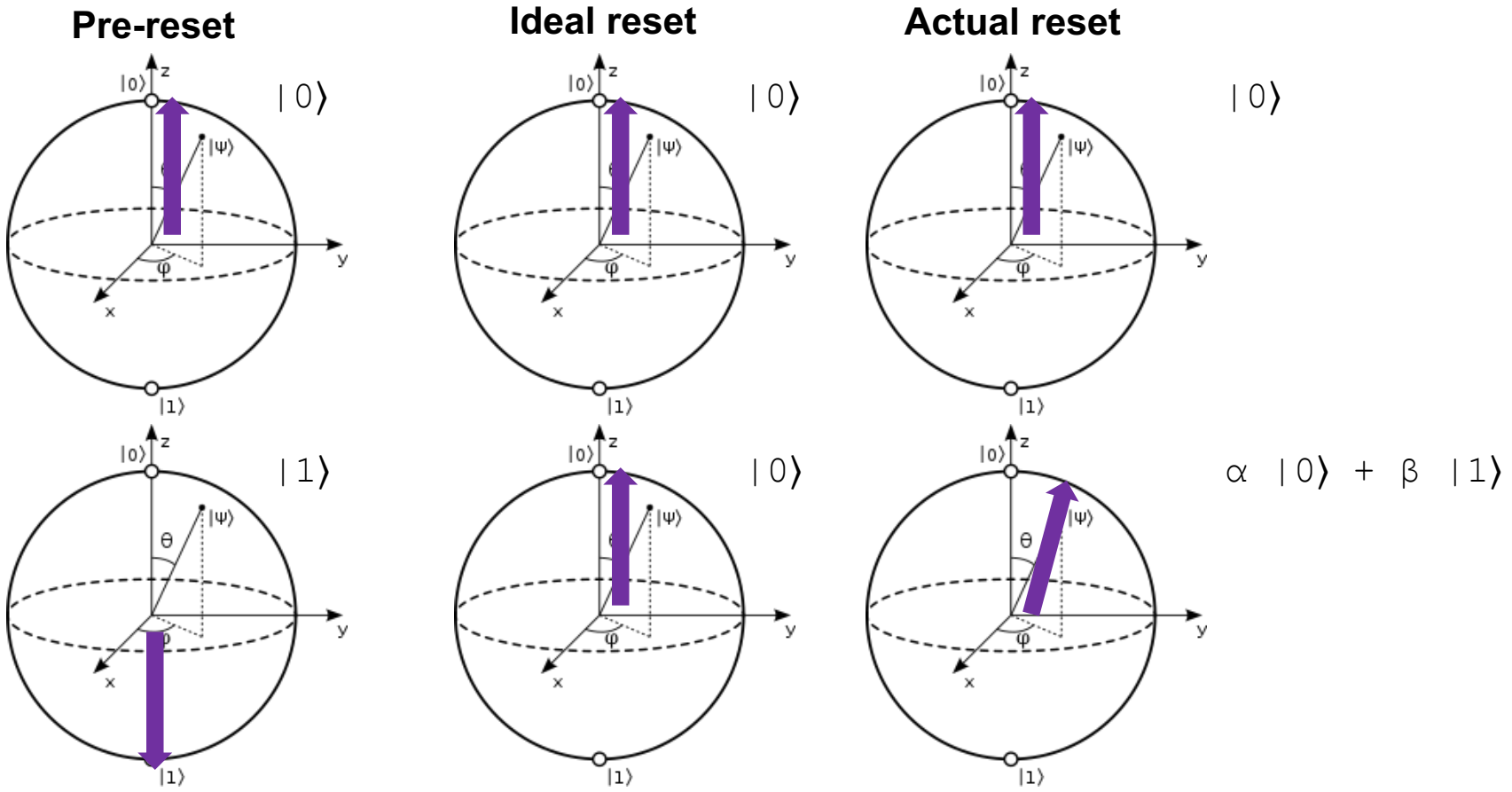


Reference:
Mi, et al. (CCS 2022)

Observed Behavior of Reset Gates

- Observed reset gate behavior on superconducting qubit quantum computers:

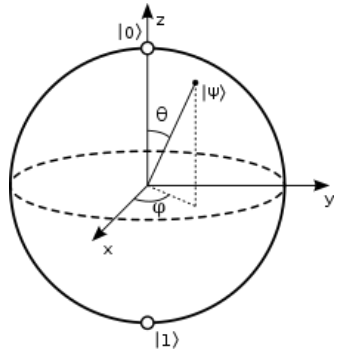




Reference:
Mi, et al. (CCS 2022)

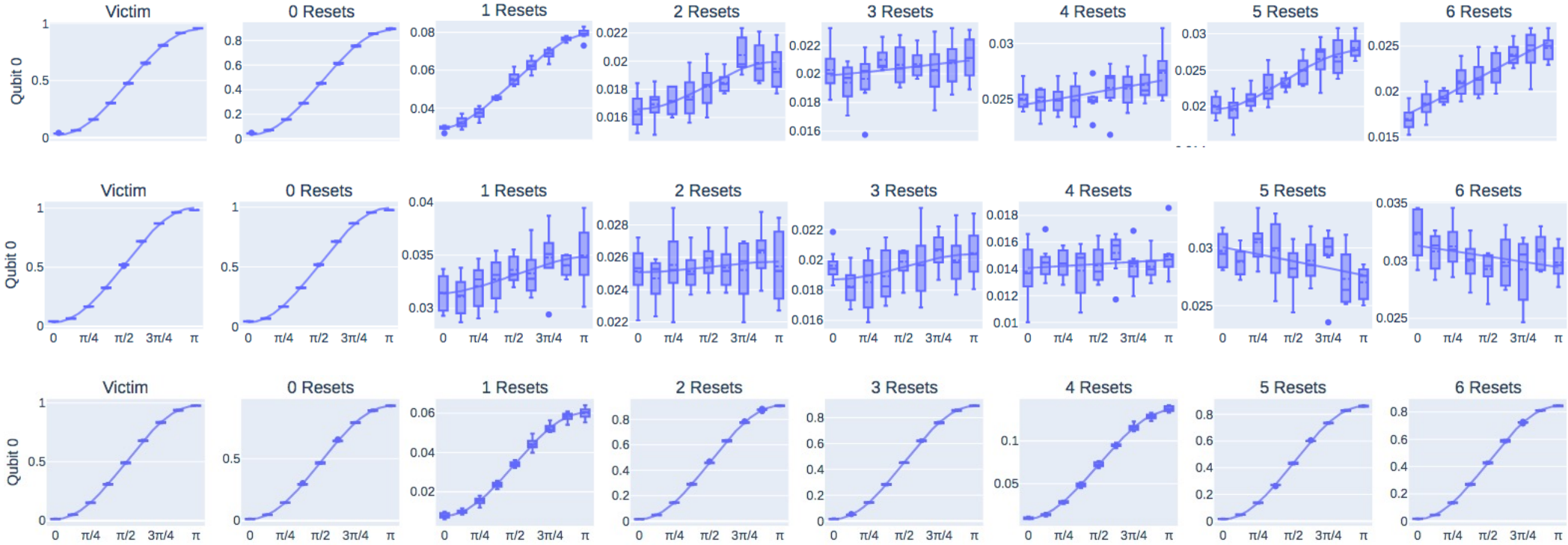
Measurements of Reset Behavior

- Experimentation demonstrated that after even multiple resets, the qubit state is not ideal $|0\rangle$
- Probability of measuring $|1\rangle$ post reset for different θ angles:



$$\alpha = \cos\frac{\theta}{2}$$

$$\beta = e^{i\varphi}\sin\frac{\theta}{2}$$



IBM Jakarta backend

IBM Lagos backend

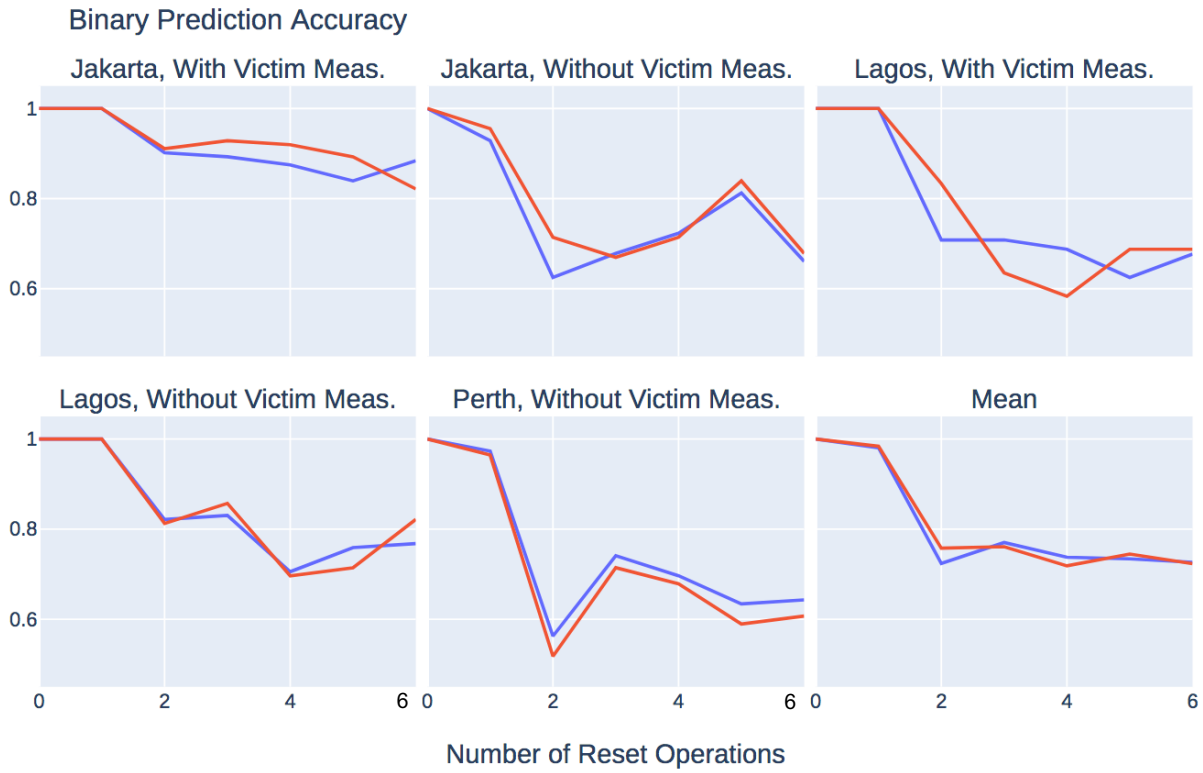
IBM Perth backend

Reference:
Mi, et al. (CCS 2022)

Automated Qubit State Prediction

- Build model that infers $|\Psi\rangle$ given measurement m and number of resets r which have occurred
- Prediction accuracy is restricted to testing data of $\theta \in \{0, \pi\}$, i.e. qubit states that approximate $|0\rangle$ or $|1\rangle$

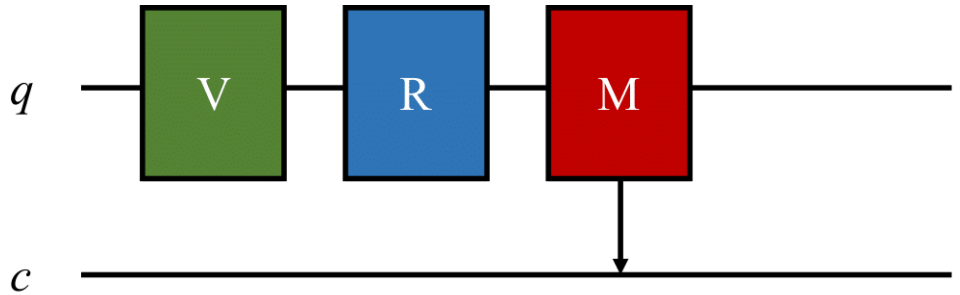
- We acquire reconstruction $\theta*$ from the channel and compare proximity of $\theta*$ to 0 and π, and choose the reconstructed qubit state to be $|0\rangle$ or $|1\rangle$ correspondingly



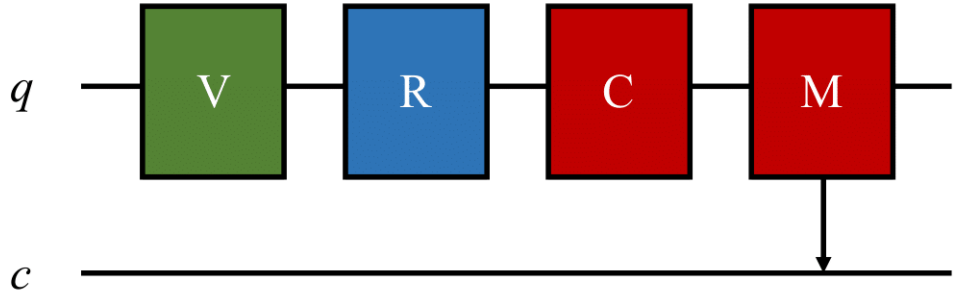
Reference:
Mi, et al. (CCS 2022)

Stealthier Reset Gate Attacks

- Reset gate attack using simple measurement can be easily detected: scan for user circuits that begin with a measurement operation
- Attackers can develop stealthier reset gate attacks to evade defenses
  - Need to explore and evaluate different strategies attackers could take, then develop defenses
- Attackers can add a masking circuit (C) before measurement to make their circuit harder to classify as a malicious circuit



Basic Attack

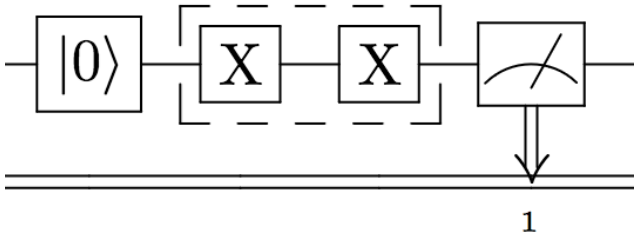


Stealthier Attack

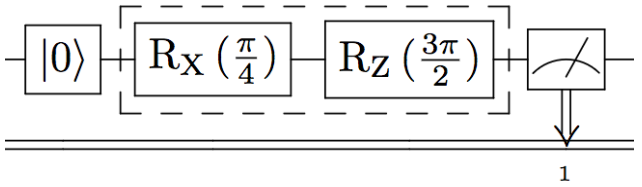
Reference:
Tan, et al. (SEED 2024)

Examples of Simple Masking Circuits

- **Identity Circuits** – Since effectively there is no rotation, the attacker's measurement should return the same values as it would be right after the reset operation
  - For example, circuits consisting of an even number of single-qubit **`X`** gates on each qubit, such that the total effective angle of rotation θ is 0



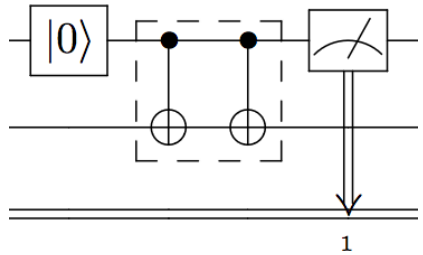
- **`RX` and `RZ` Gate Circuits** – Because the rotation angle is known, the attacker can infer the qubit 1-output probabilities as they would be right after the reset gate
  - For example, circuits consisting of single-qubit gates with effective θ (**`RX`** gate) rotation and φ (**`RZ`** gate) rotation

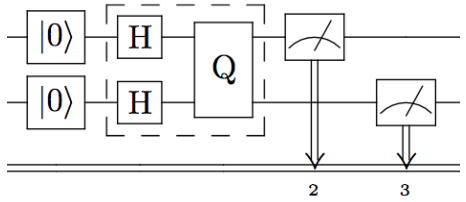


Reference:
Tan, et al. (SEED 2024)

Examples of Simple Masking Circuits

- **`cx` Gate Circuits** – The control qubits of `cx` gate experience delay (due to duration `cx` gate) but otherwise can be leveraged by an attacker since they do not experience any rotations
  - For example, circuits consisting of multiple `cx` gates
- **QASM Benchmark Circuits** – Realistic circuits would be ideal masking circuits, but will make inferring the victim's state most difficult if not impossible
  - For example, circuits from the QASM benchmark suite

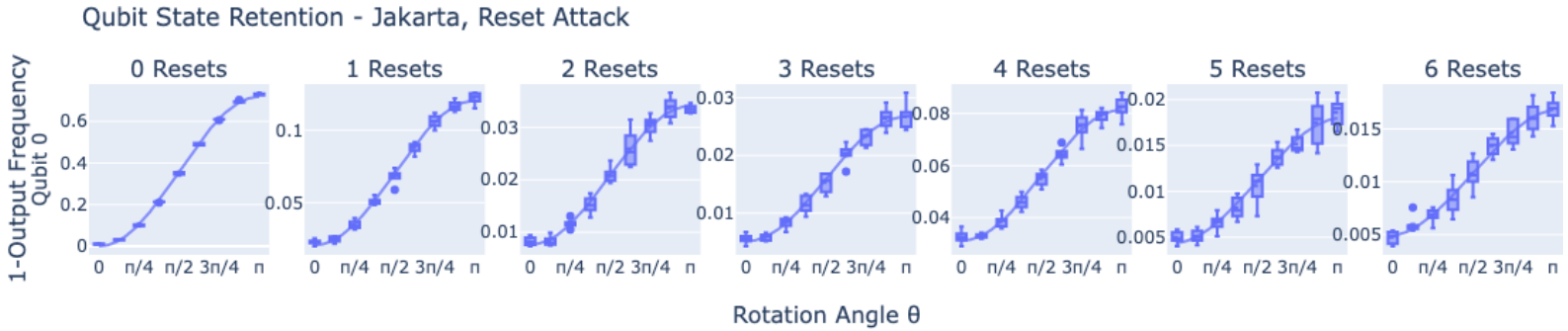




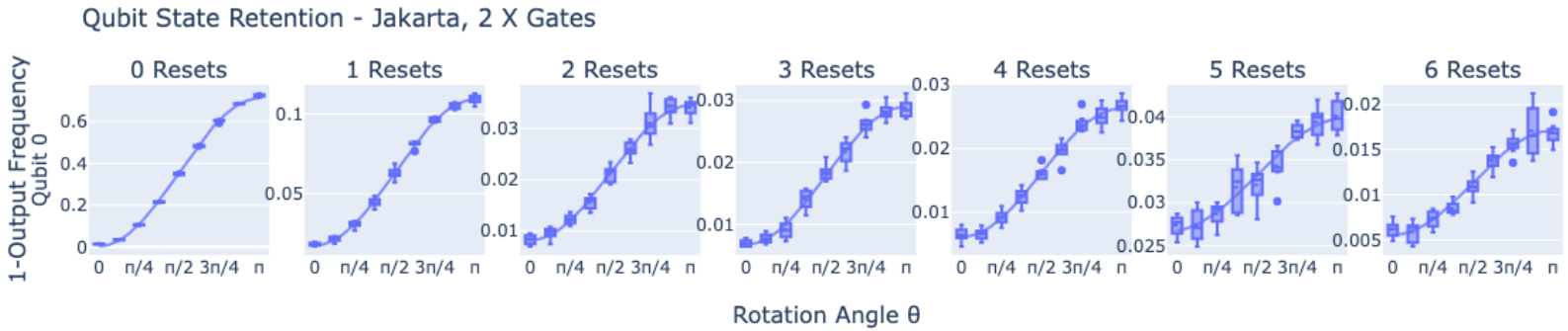
Reference:
Tan, et al. (SEED 2024)

Recovering Victim's State when Masking Circuit is Used

- Comparison to baseline when no masking circuit is used:



- Probability of measuring $|1\rangle$ post reset for different θ angles when **identity circuit** is used:

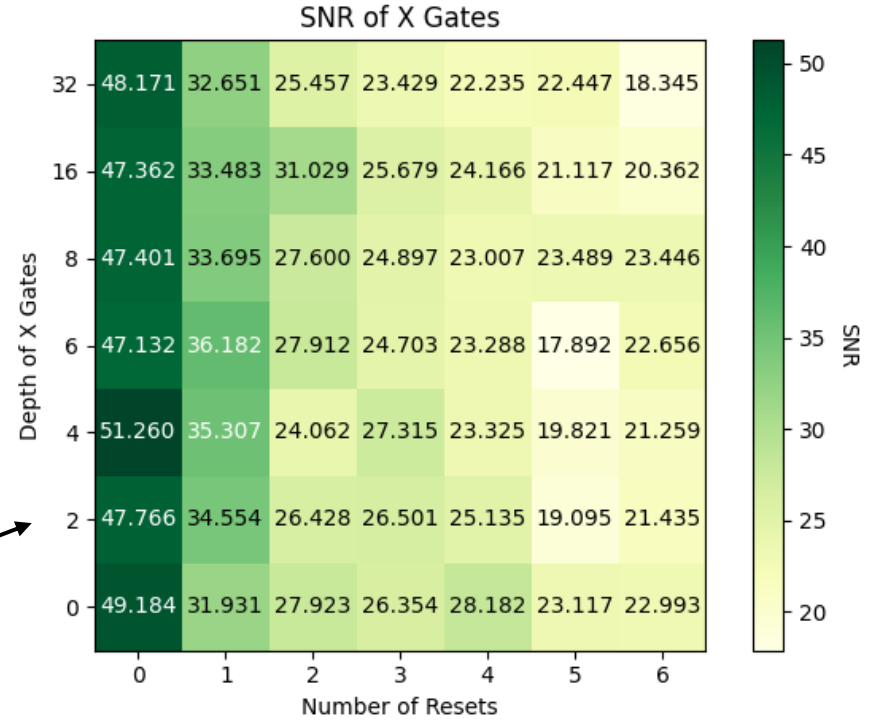


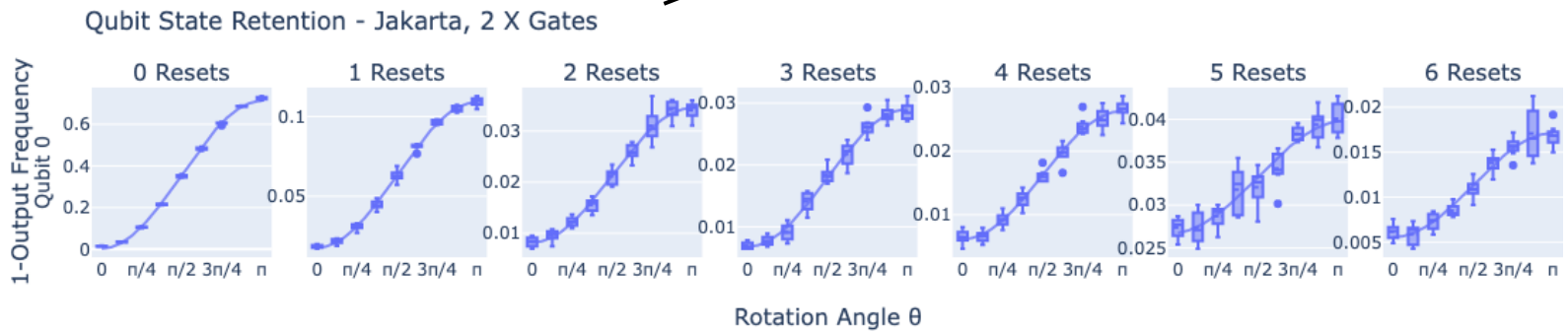
Reference:
Tan, et al. (SEED 2024)

Analysis of Masking Circuit Effectiveness with SNR

- The signal-to-noise (SNR) ratio is defined as $\mathbf{SNR = a/\sigma}$
  - $\mathbf{a}$ is the error channel characterization parameter, which represents the amplitude of the sigmoid fit
  - compute the standard deviation in 1-output frequency for each fixed θ as φ varies and then compute the average standard deviations over all input θ values which gives σ

- **Identity circuit** example:





Reference:
Tan, et al. (SEED 2024)

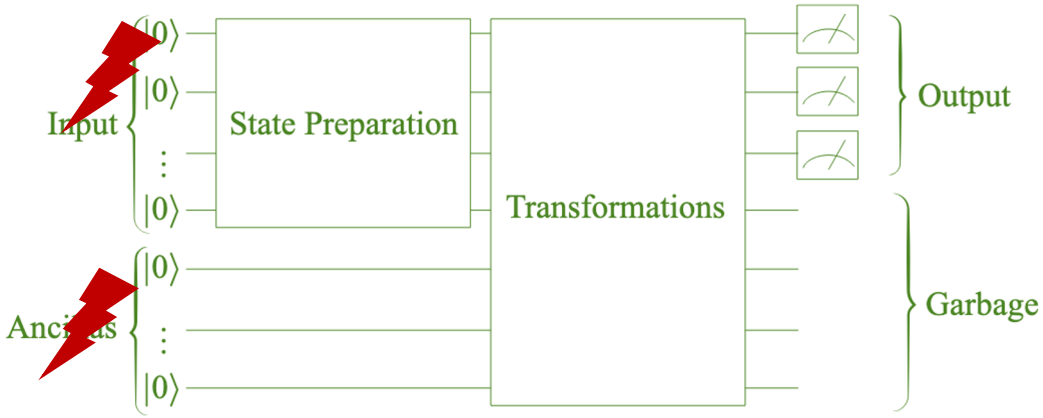
Attacks on Qubit Outputs Summary and Outlook

Many attacks have been demonstrated:

- Reset gate attack (2022)
- More stealthy reset gate attack (2024)

CA1.A: Attacks on initial input qubits states

CA1.B: Attacks on initial ancilla qubits states



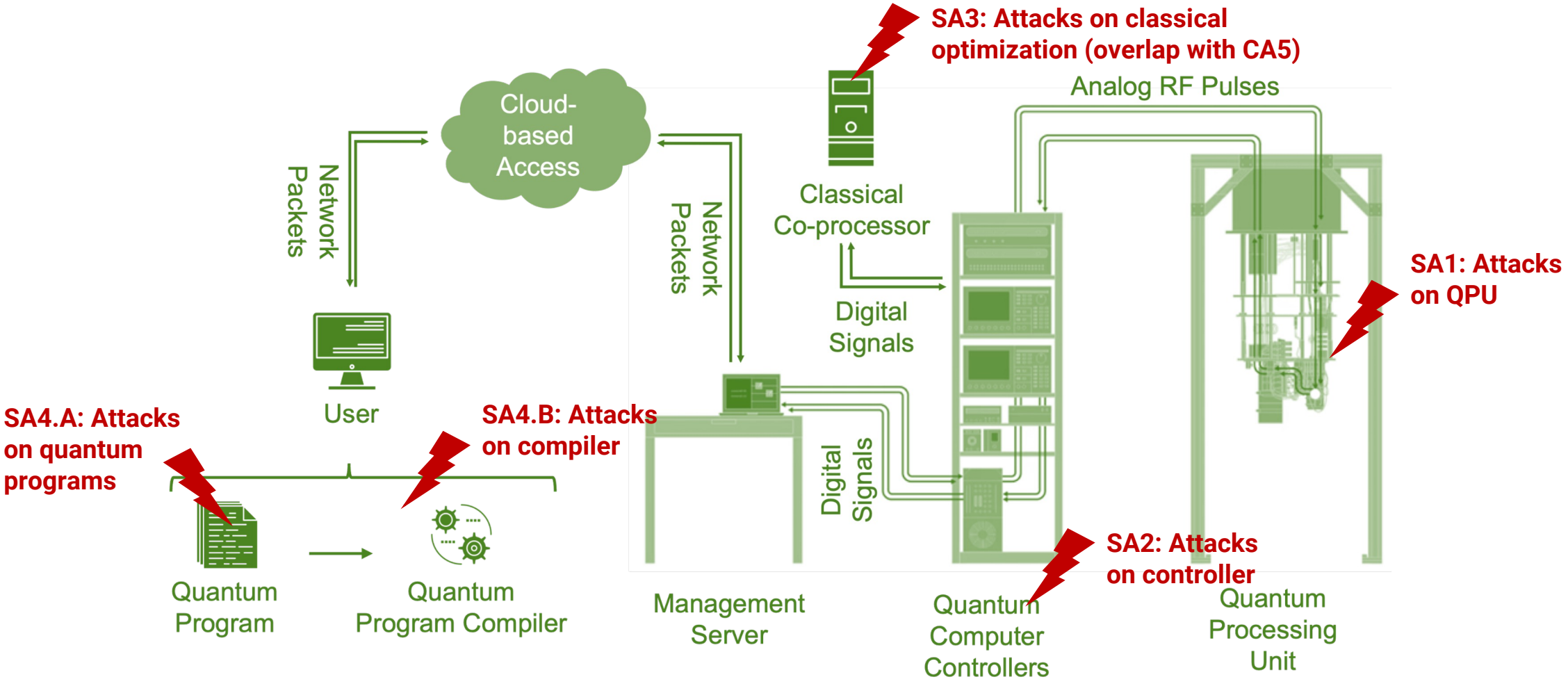
Various defenses have been proposed in parallel:

- Letting qubits fully thermalize
- Apply one time pad and randomly flip qubits after measurement

Future directions in this research:

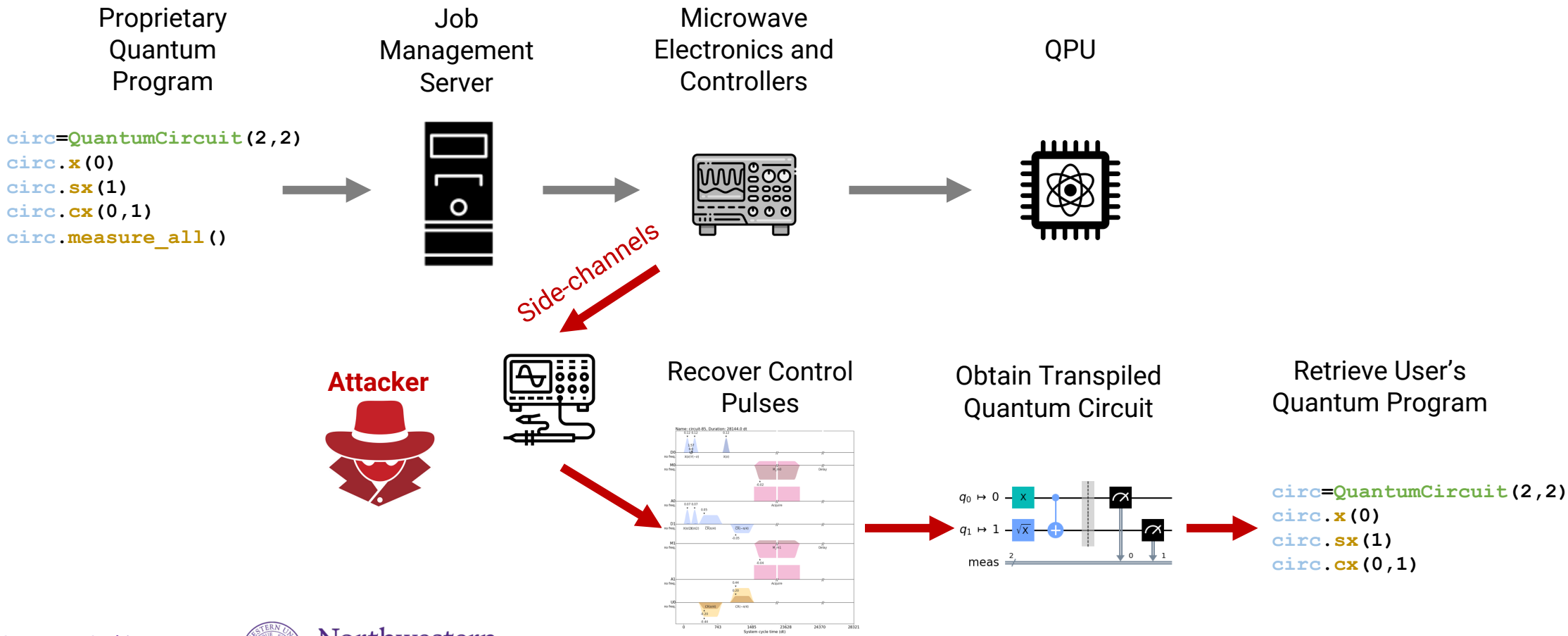
- Reset mechanisms in other superconducting architectures

Side Channels in Controller



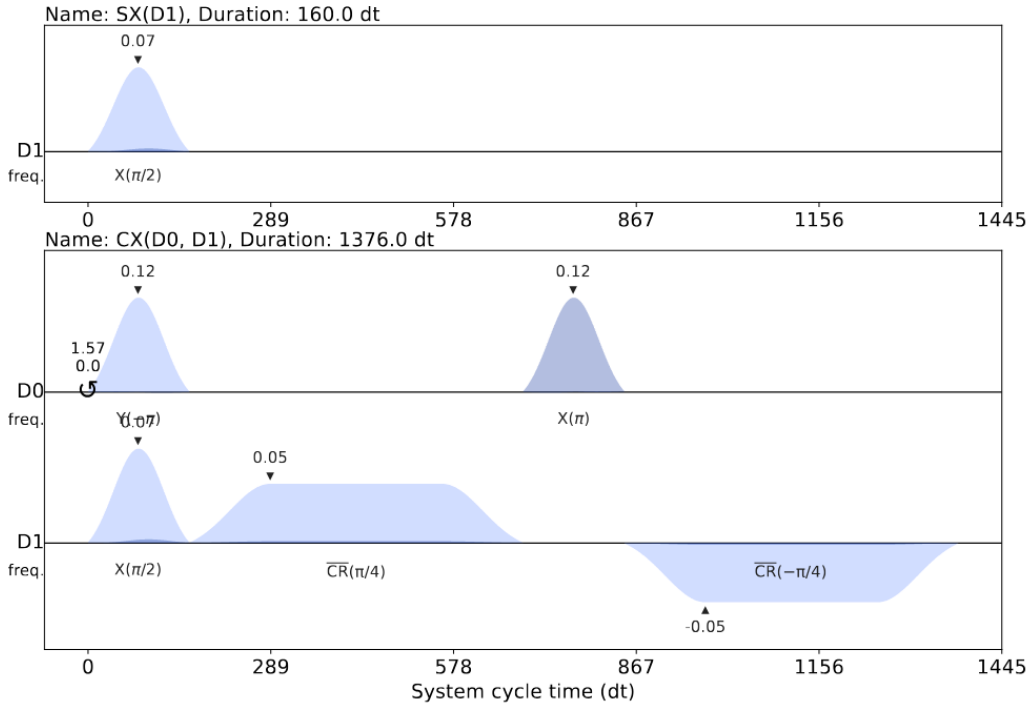
Information Leaks from Controllers

- The side-channel attack setup assumes access to collect side-channel information from the controller electronics:



Single- and Two-Qubit Control Pulses

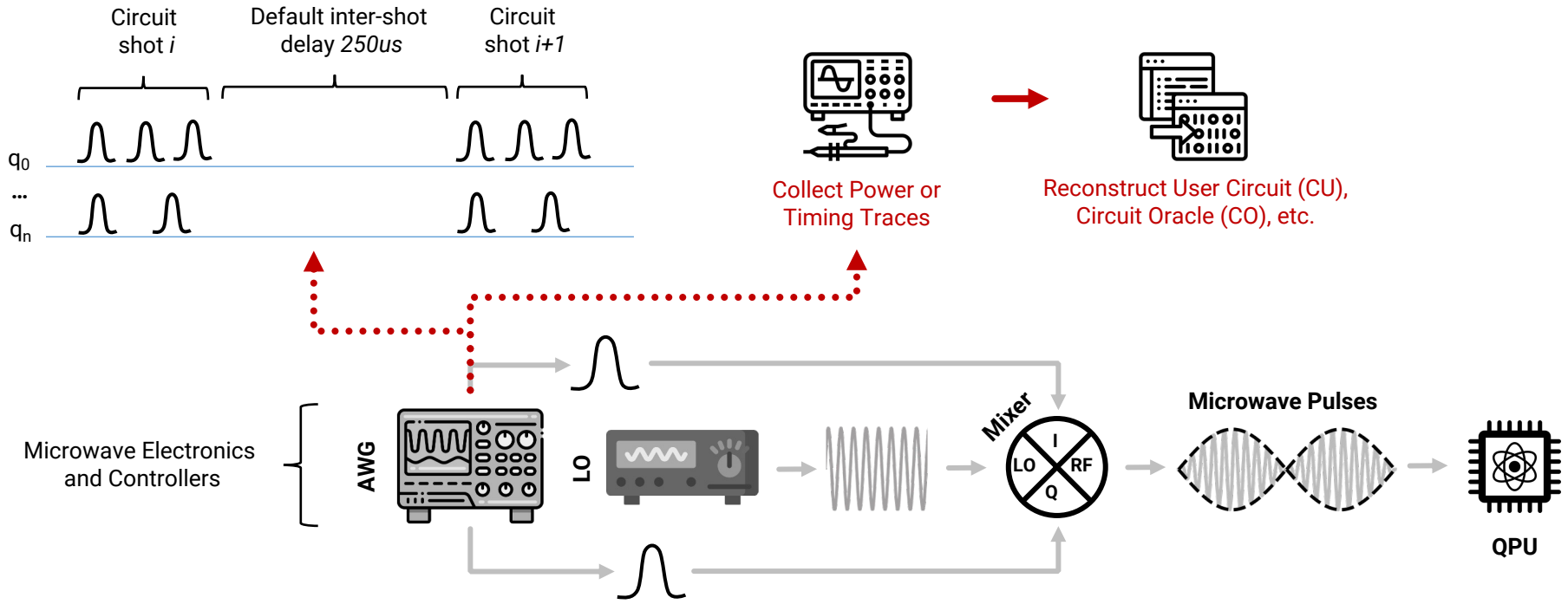
- Control pulses are defined by frequency, duration, and envelope (shape):



- Each qubit has slightly different frequency and other parameters

Possible Setup for Information Leak via Side Channels

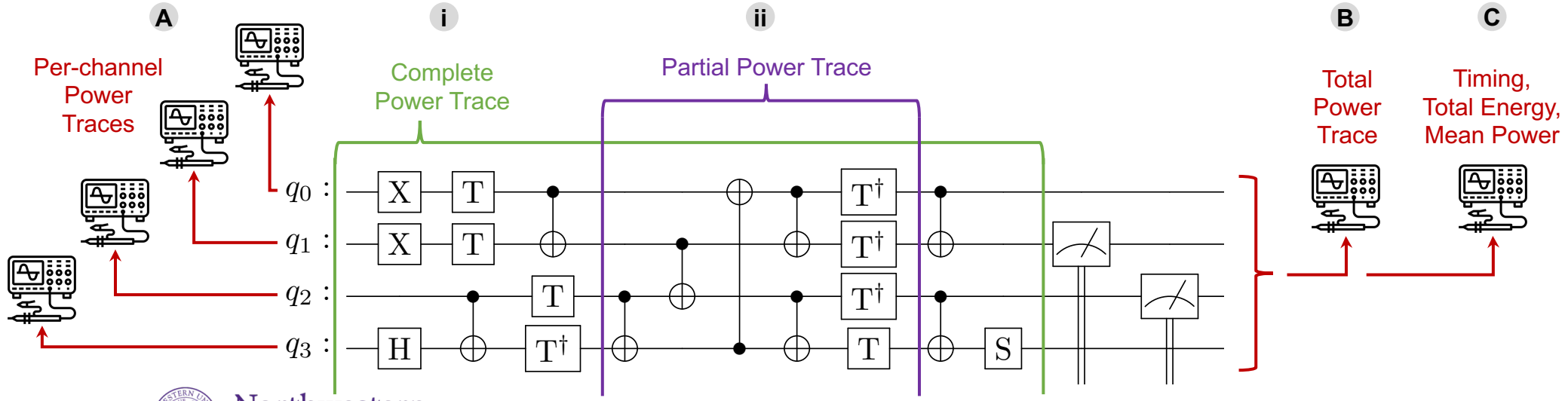
- Key target for the side-channel attacks are the arbitrary waveform generators (AWGs):



Side Channel Taxonomy for Controller Attacks

- We can consider different capabilities of the attackers:
  A. Can measure per-channel power traces (strongest)
  B. Can measure total power trace for all channels
  C. Can measure scalar quantities such as timing, total energy, or mean power (weakest)

- Attacker can target different parts of the circuit:
  i. Complete circuit
  ii. Portion of the circuit



Attacker Goals for Information Leak

- **(UC) User Circuit Identification –** Given knowledge about the set of possible circuits executed on the quantum computer, find which circuits the user actually executed
- **(CO) Circuit Oracle Identification –** Given a known circuit, such as Bernstein-Vazirani, but an unknown oracle, find the configuration of the oracle used in that circuit
- **(CA) Circuit Ansatz Identification –** Given a known circuit, such as a variational circuit used in machine learning applications, but an unknown ansatz, find the configuration of the ansatz used in that circuit
- **(QM) Qubit Mapping Identification –** Given a known circuit, identify the placement of which physical qubits were used
- **(QP) Quantum Proc. Identification –** Given knowledge about the pulses for quantum processors and a circuit, find the quantum processor on which the circuit was executed
- **(CR) Circuit Reconstruction –** Given knowledge about the pulses for quantum computer basis gates, reconstruct the complete, unknown circuit from the power traces

Types of Side Channels

C

Timing Attack – Measure execution time of one shot of a circuit to recover user circuits (**UC**)

Total Energy Attack – Measure total energy consumption of the control equipment over execution of one shot of a circuit to recover users' circuits (**UC**)

Mean Power Attack – Measure mean power consumption of the control equipment over execution of one shot of a circuit to recover users' circuits (**UC**)

B

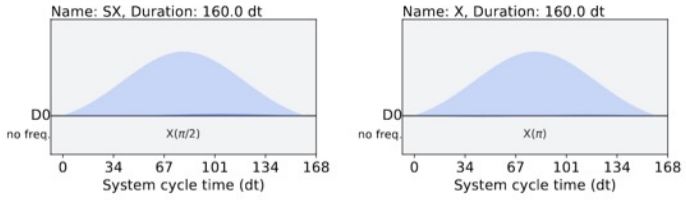
Total Power Trace Attack – Measure a trace of the total power consumption of the control equipment over execution of one shot of a circuit of all the channels to recover user circuits (**UC**), circuit oracle (**CO**), circuit ansatz (**CA**), qubit mapping (**QM**), and quantum processor (**QP**) with some accuracy

A

Per-Channel Power Trace Attack – Measure a traces of the power consumption of the control equipment over execution of one shot of a circuit of each channels to perform circuit reconstruction (**CR**), thus recovering user circuits

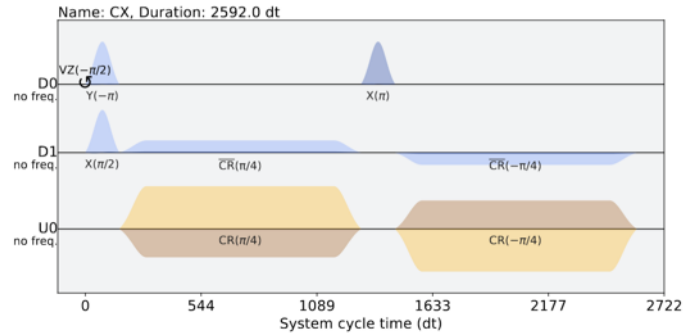
Simulation of Attacks from Pulse Information

- Control pulse information (amplitude and duration) can be obtained from quantum computer provider or vendor
  - First order assumption power consumption proportional to the pulse amplitude and duration
  - More detailed models of power consumption can be developed

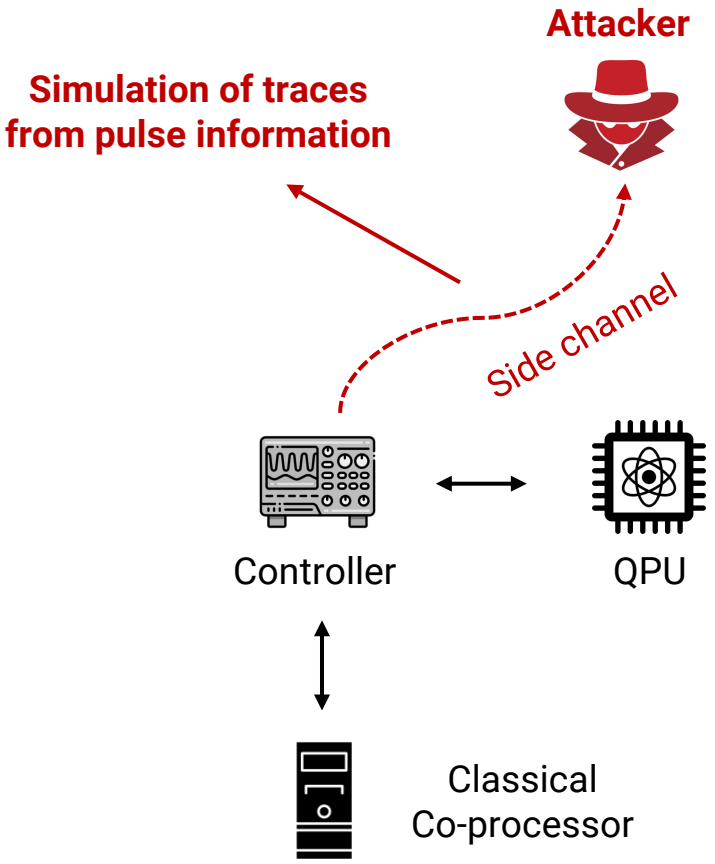


SX pulse

X pulse



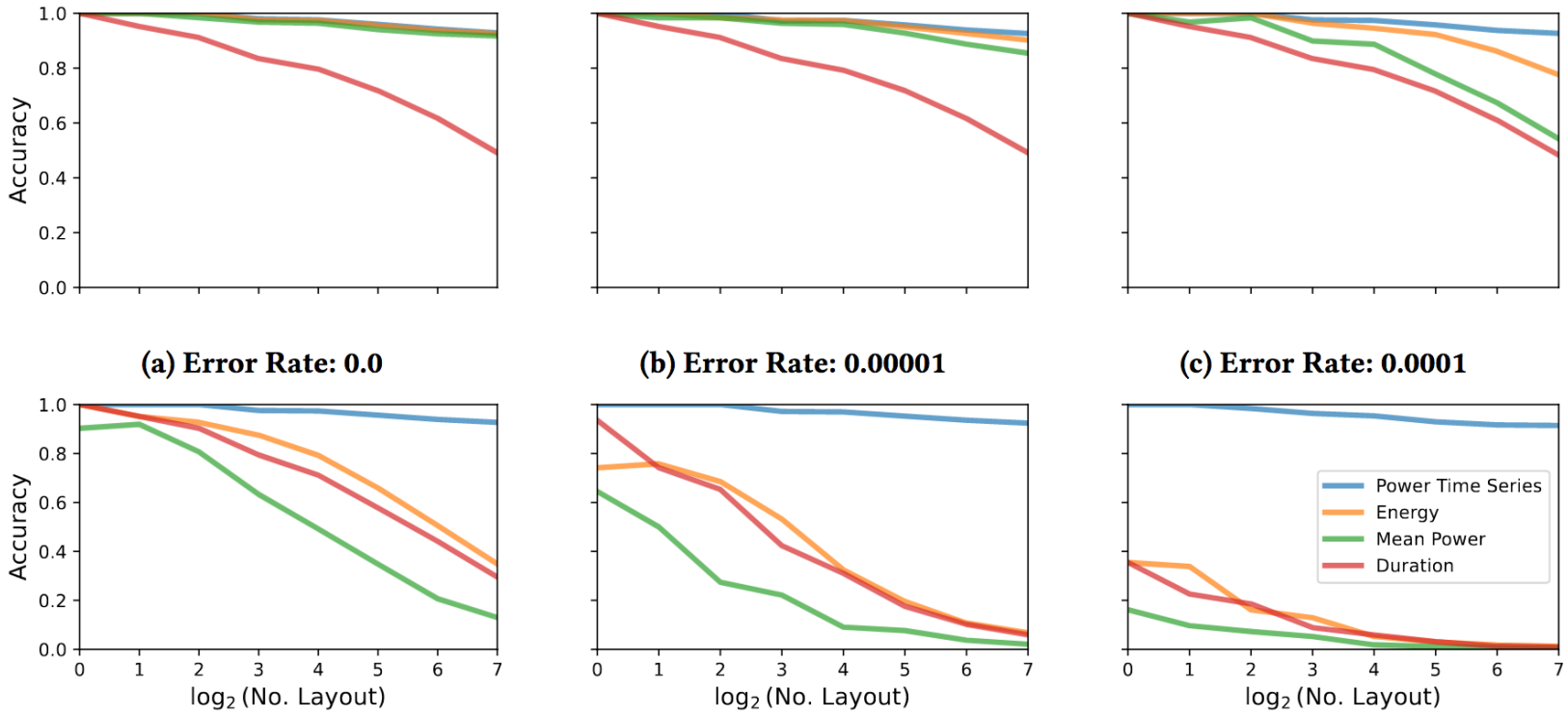
CX pulse



Example User Circuit (UC) Identification

- UC attacks can be carried out by collecting energy, mean power or duration information, as well as by power time series collection
  - Testing done using QASMBench ver. 1.4

- Goal to identify 1 out of N benchmark circuits:



(a) Error Rate: 0.0 (b) Error Rate: 0.00001 (c) Error Rate: 0.0001

(d) Error Rate: 0.001 (e) Error Rate: 0.01 (f) Error Rate: 0.1

Example Circuit Oracle (CO) Identification

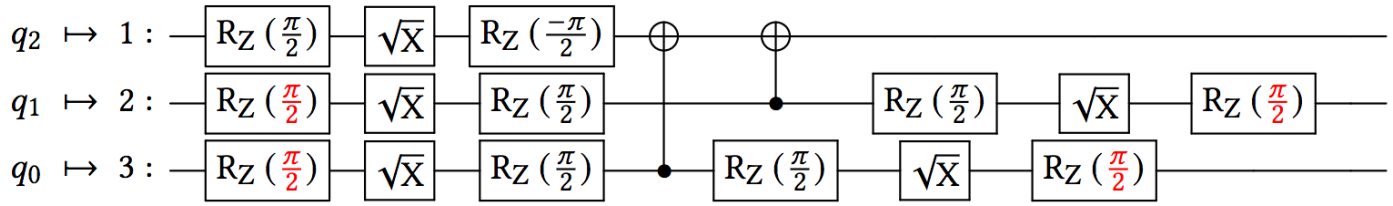
- Many quantum algorithms consist of oracles, which act like black boxes that return desired quantum states based on the input

- Normalized circuit distance for Bernstein-Vazirani (BV), Deutsch-Jozsa (DJ), and Grover's Search (GS) with the number of qubits from 1 to 6 on `ibm_lagos`:

Algorithm	Number of Qubits/Oracles					
	1/2	2/4	3/8	4/16	5/32	6/64
Bernstein-Vazirani	1.00	0.30	0.07	0.06	0.07	0.06
Deutsch-Jozsa	0.00	0.00	0.00	0.00	0.00	0.00
Grover's Search	0.00	0.00	0.00	0.00	0.00	0.00

  - For BV, since the oracles are quite different from each other, the minimum circuit distance is not 0, which means the oracles **can** be distinguished from each other

  - For DJ and GS, the circuits for different oracles can be the same, and the only changes are the angles of the rotation gates, such as RZ gate, which means the oracles **cannot** be distinguished



Example Qubit Mapping (QM) and Quantum Proc. (QP) Ident.

- Because of the unique per qubit control pulses, the power traces also encode the information of the physical qubits to which the quantum gates are applied to, i.e. QM
- Because of the unique per qubit control pulses, the power traces also encode the information of the quantum processor on which the circuit executes, i.e. QP
- The minimum normalized circuit distance is used to evaluate the results for QM and QP, the larger value means it is simpler to distinguish the circuits:

QASMBench Benchmark	Parameters			Attacks		
	Qubit	Gate	CX	QM	QP	CR
deutsch	2	10	1	0.025	0.116	✓
dnn	2	306	42	0.039	0.116	✓
grover	2	15	2	0.143	0.116	✓
iswap	2	14	2	0.143	0.116	✓
quantumwalks	2	38	3	0.125	0.117	✓
basis_change	3	85	10	0.673	0.068	✓
fredkin	3	31	17	0.800	0.411	✓
linearsolver	3	26	4	0.735	0.080	✓
qaoa	3	35	9	0.546	0.570	✓
teleportation	3	12	2	0.473	0.075	✓
toffoli	3	24	9	0.096	0.573	✓
wstate	3	47	21	0.789	0.101	✓
adder	4	33	16	0.727	0.201	✓
basis_trotter	4	2353	582	0.895	0.220	✓
bell	4	53	7	0.781	0.196	✓
cat_state	4	6	3	0.744	0.241	✓
hs4	4	28	4	0.545	0.327	✓
inverseqft	4	30	0	0.000	0.001	✓
qft	4	50	18	0.817	0.287	✓
qrng	4	12	0	0.000	0.001	✓
variational	4	58	16	0.792	0.239	✓
vqe	4	73	9	0.660	0.194	✓
vqe_uccsd	4	238	88	0.858	0.241	✓
error_c3	5	249	61	0.855	0.220	✓
lpn	5	17	2	0.576	0.194	✓
pea	5	126	57	0.874	0.210	✓
qec_en	5	52	16	0.746	0.250	✓
qec_sm	5	8	4	0.573	0.266	✓
qaoa	6	408	84	0.869	0.283	✓
simon	6	65	23	0.796	0.605	✓
vqe_uccsd	6	2289	1199	0.906	0.278	✓
hhl	7	1092	298	0.873	0.317	✓

Side Channels in Controller Summary and Outlook

Many attacks have been demonstrated:

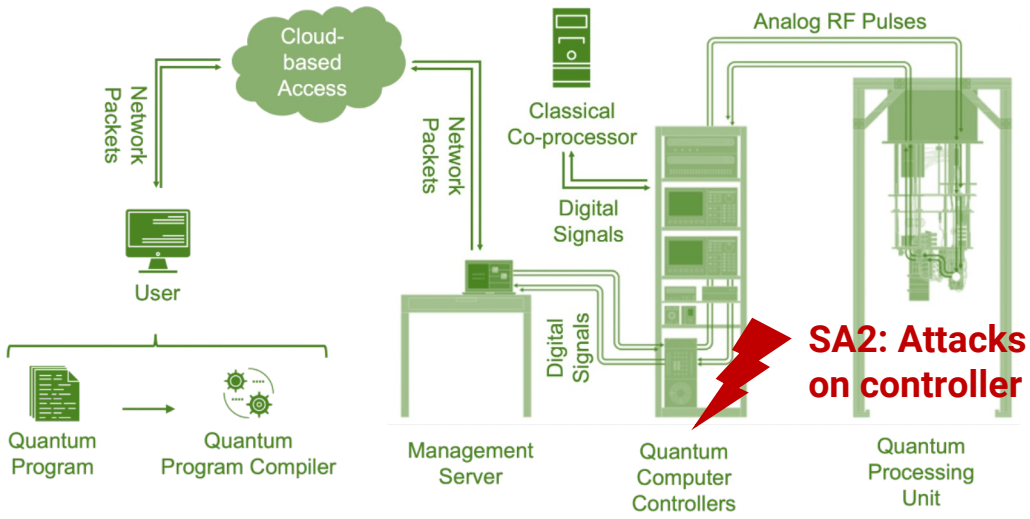
- Side channel attack on controller (since about 2023)

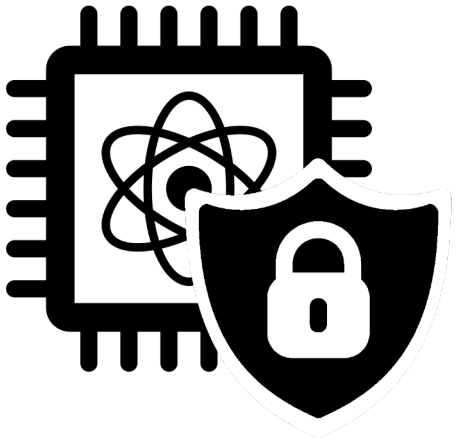
Various defenses have been proposed in parallel:

- Add gates or change duration of circuits
- Obfuscate circuits assuming trusted fridge (2024)

Future directions in this research:

- Better modeling of controller and power related to control pulses
- Attacks on virtual gates





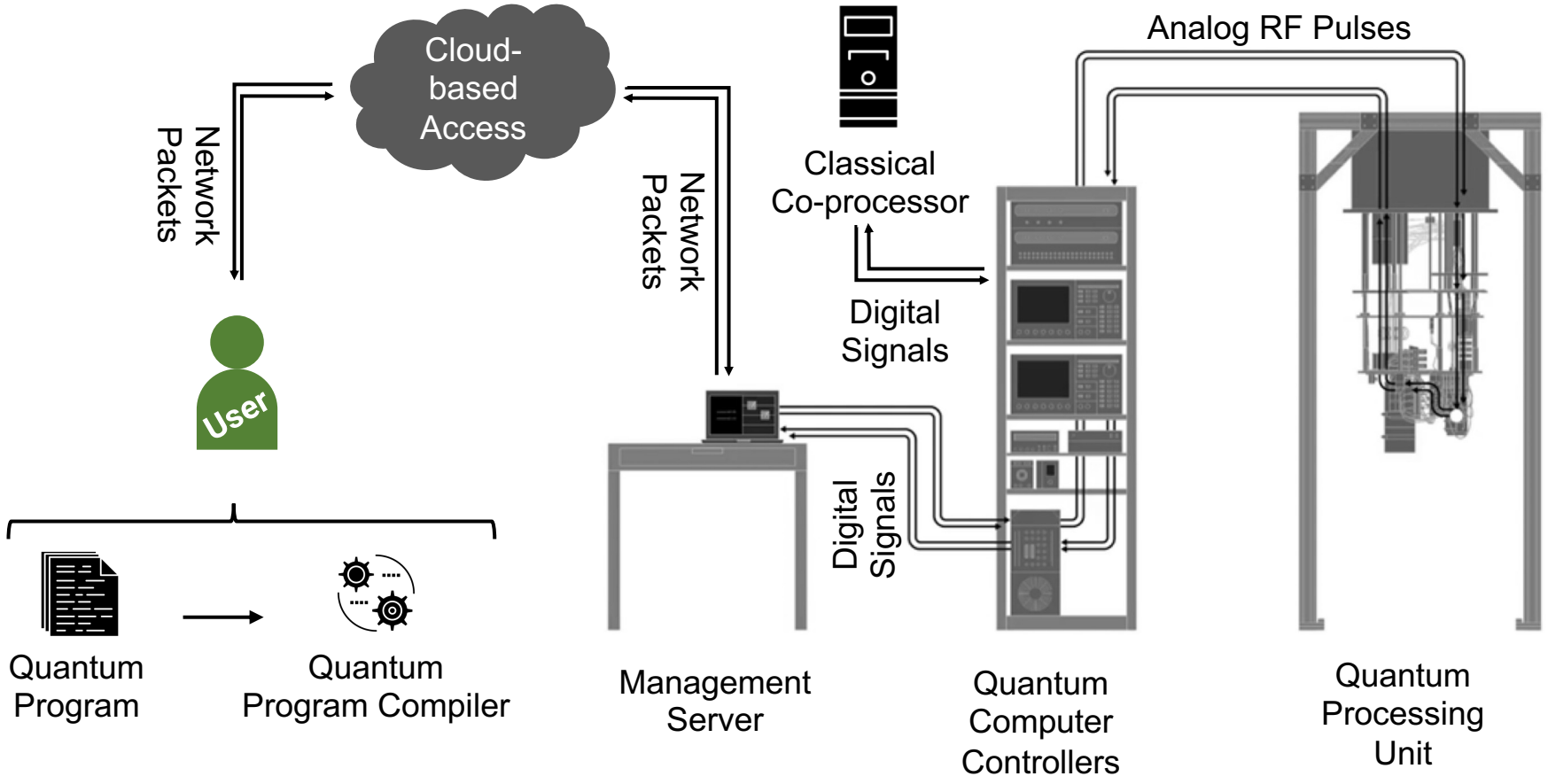
Tutorial on Security of Quantum Computing Systems

Trusted Execution Environments for NISQ Systems

Computer Architecture and Security Lab (CASLAB)

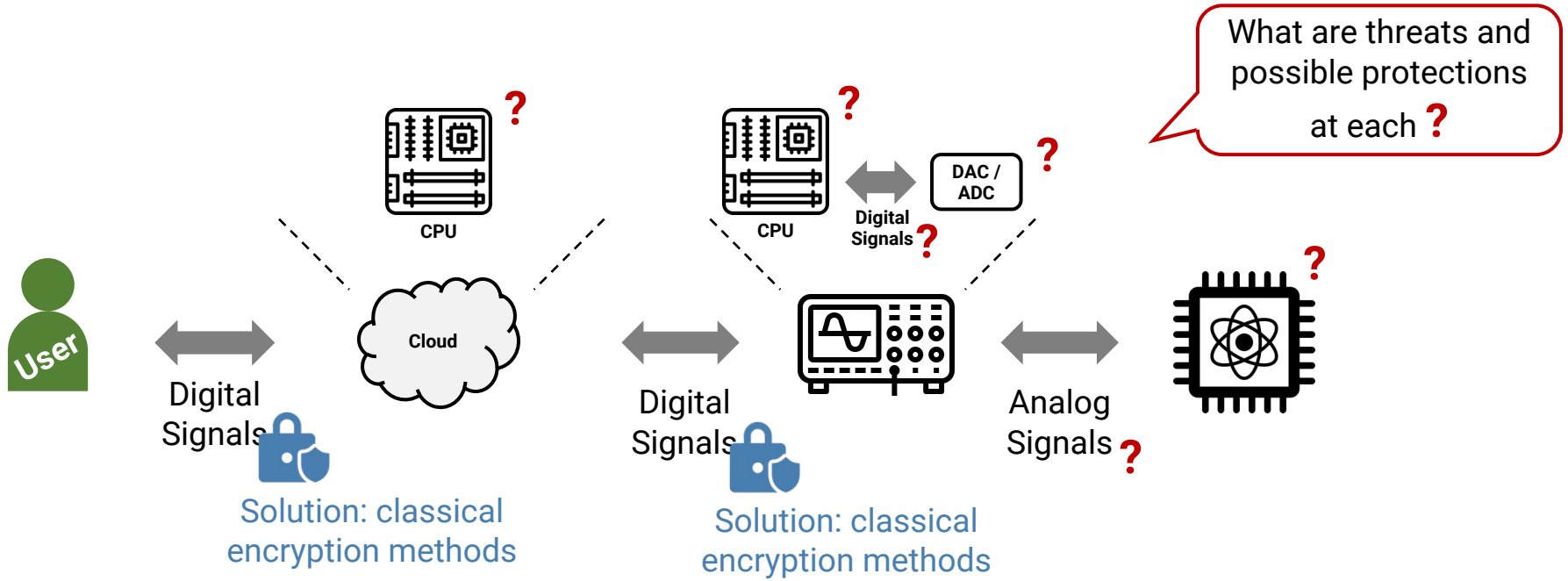
Northwestern University

Typical Quantum Computer System Overview



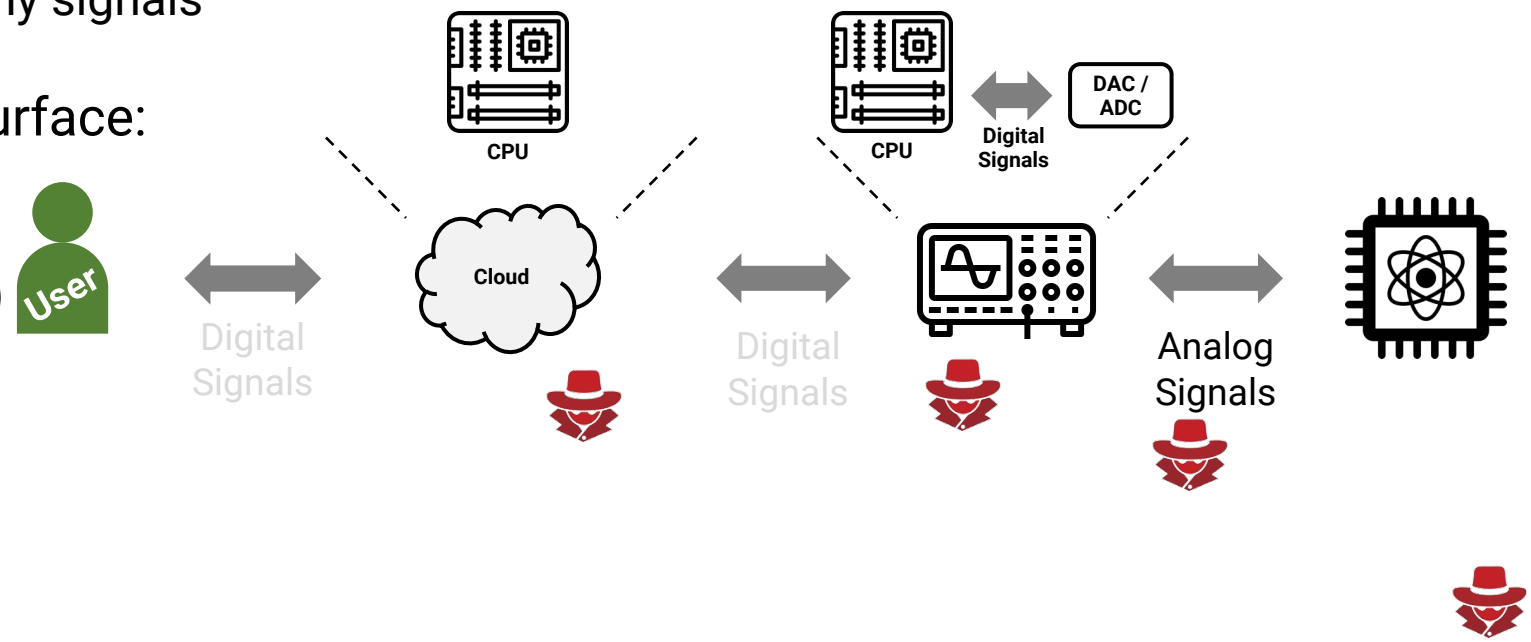
Typical Quantum Computer System Overview

- Abstract view of the quantum computer system can aid in analyzing the potential security threats:



Assumptions and Threats

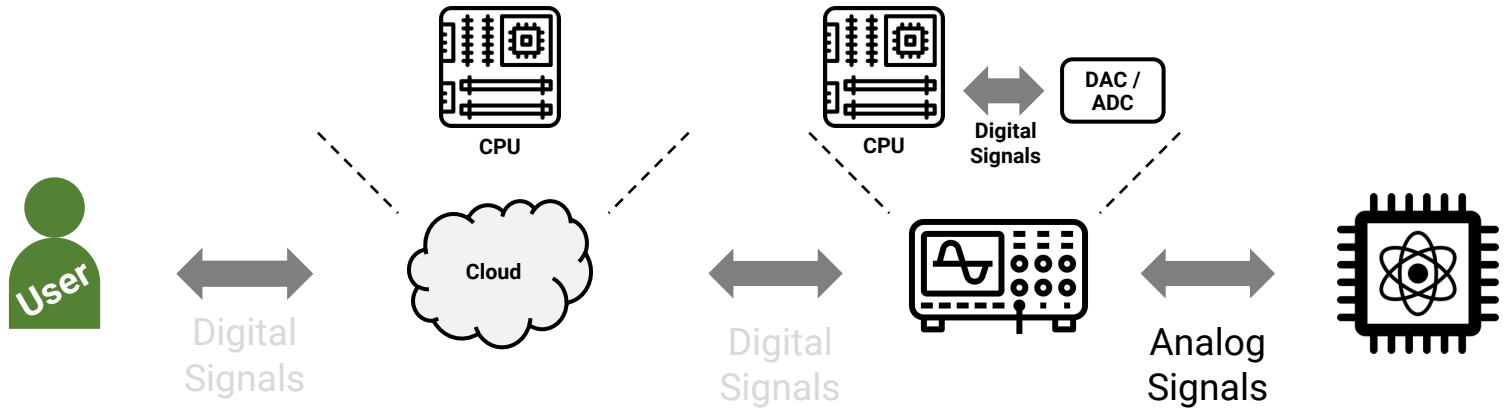
- Assume honest-but-curious attackers
  - Do not inject errors or manipulate signals
  - Only eavesdrop on operation of the quantum computer and any signals
- Resulting attack surface:
  - Cloud (CPU)
  - Controller (CPU)
  - Controller (DAC)
  - QPU



Assumptions and Threats

Solution: Blind Quantum Computation →

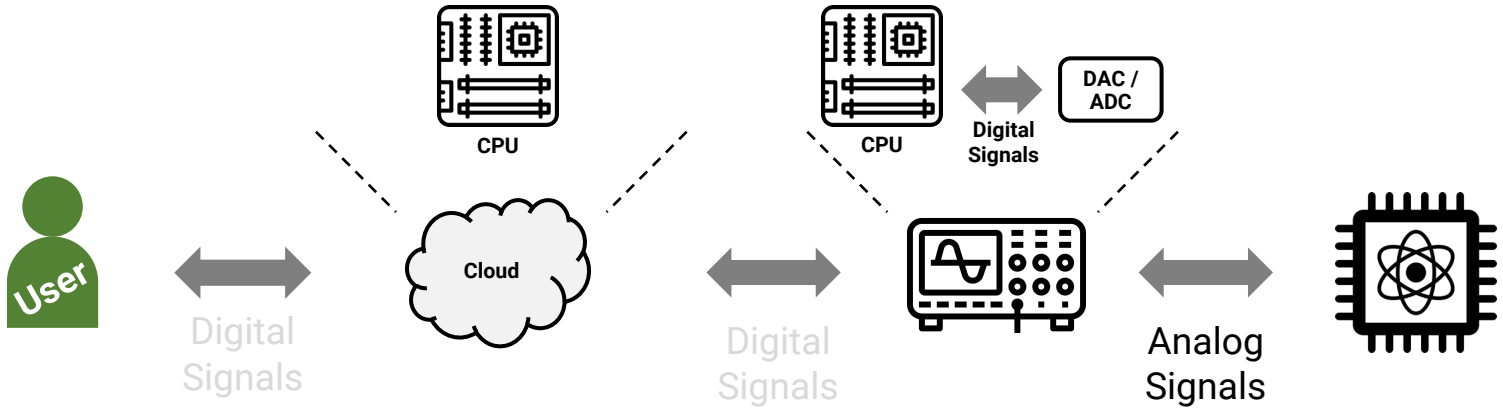
	Cloud (CPU)	Controller (CPU)	Controller (DAC)	QPU	Cloud (CPU)	Controller (CPU)	Controller (DAC)	QPU	
	Untrusted	Untrusted	Untrusted	Untrusted	Trusted	Untrusted	Untrusted	Untrusted	?
?	Untrusted	Untrusted	Untrusted	Trusted	Trusted	Untrusted	Untrusted	Trusted	?
?	Untrusted	Untrusted	Trusted	Untrusted	Trusted	Untrusted	Trusted	Untrusted	?
?	Untrusted	Untrusted	Trusted	Trusted	Trusted	Untrusted	Trusted	Trusted	?
?	Untrusted	Trusted	Untrusted	Untrusted	Trusted	Trusted	Untrusted	Untrusted	?
?	Untrusted	Trusted	Untrusted	Trusted	Trusted	Trusted	Untrusted	Trusted	?
?	Untrusted	Trusted	Trusted	Untrusted	Trusted	Trusted	Trusted	Untrusted	?
?	Untrusted	Trusted	Trusted	Trusted	Trusted	Trusted	Trusted	Trusted	?



Assumptions and Threats

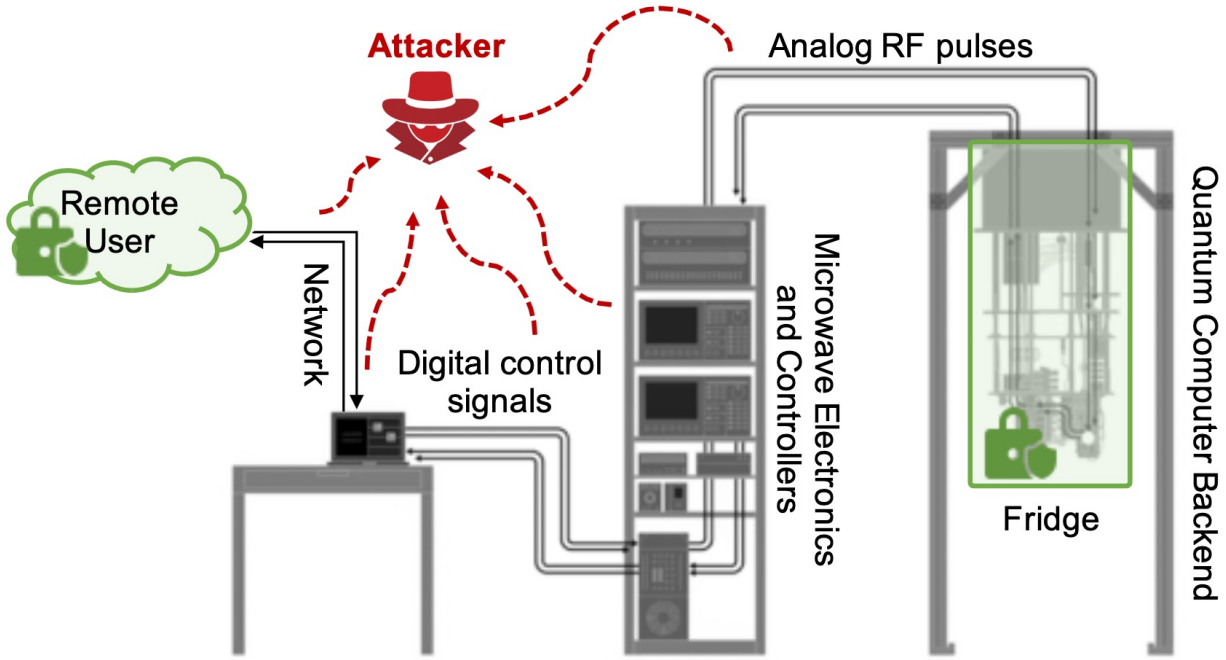
New Solutions:
QC-TEE & CASQUE

	Cloud (CPU)	Controller (CPU)	Controller (DAC)	QPU	Cloud (CPU)	Controller (CPU)	Controller (DAC)	QPU	
	Untrusted	Untrusted	Untrusted	Untrusted	Trusted	Untrusted	Untrusted	Untrusted	?
	Untrusted	Untrusted	Untrusted	Trusted	Trusted	Untrusted	Untrusted	Trusted	?
?	Untrusted	Untrusted	Trusted	Untrusted	Trusted	Untrusted	Trusted	Untrusted	?
?	Untrusted	Untrusted	Trusted	Trusted	Trusted	Untrusted	Trusted	Trusted	?
?	Untrusted	Trusted	Untrusted	Untrusted	Trusted	Trusted	Untrusted	Untrusted	?
?	Untrusted	Trusted	Untrusted	Trusted	Trusted	Trusted	Untrusted	Trusted	?
?	Untrusted	Trusted	Trusted	Untrusted	Trusted	Trusted	Trusted	Untrusted	?
?	Untrusted	Trusted	Trusted	Trusted	Trusted	Trusted	Trusted	Trusted	?



QC-TEE Assumptions

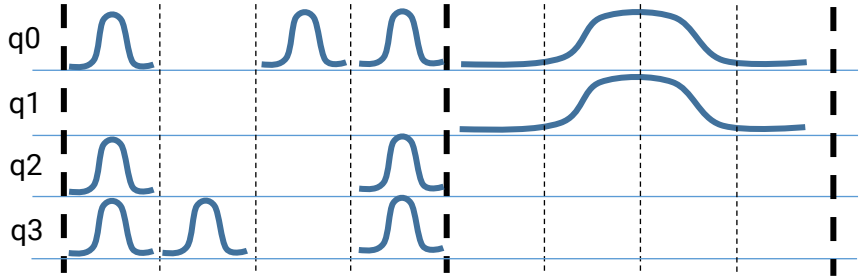
- Focus on superconducting qubit quantum computers
- Fridge of superconducting qubit quantum computers can form natural trust boundary
  - Temperature and pressure changes can be easily detected when fridge is opened
- **Key idea:** need to protect or obfuscate the analog RF pulses going into the (trusted) QPU
  - Digital signals (going through untrusted cloud or controller CPU) are equivalent to the RF pulses
  - Assume we cannot put signal generators or other large electronics in the fridge



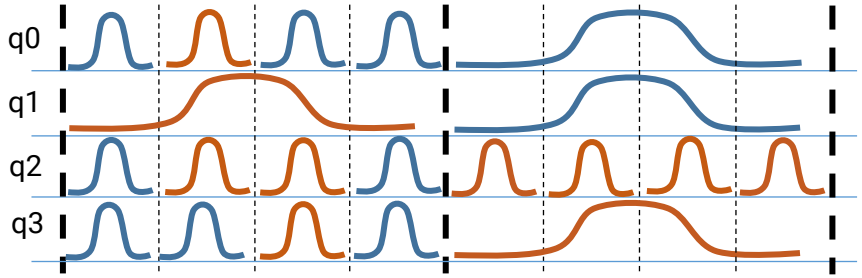
High-Level Idea of Hiding Control Pulses

- Analog control pulses cannot be ”encrypted” unlike digital information, but we can hide or obfuscate them instead

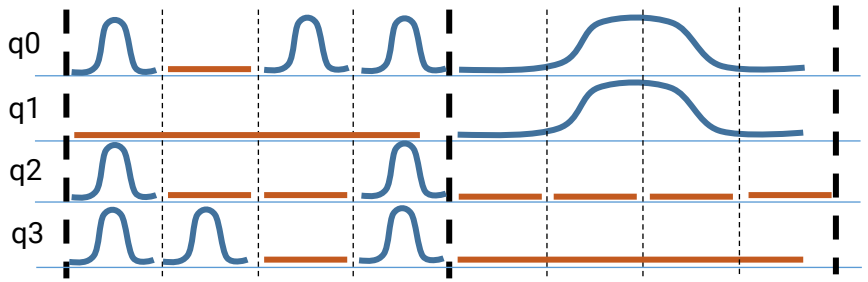
1. Input user circuit:



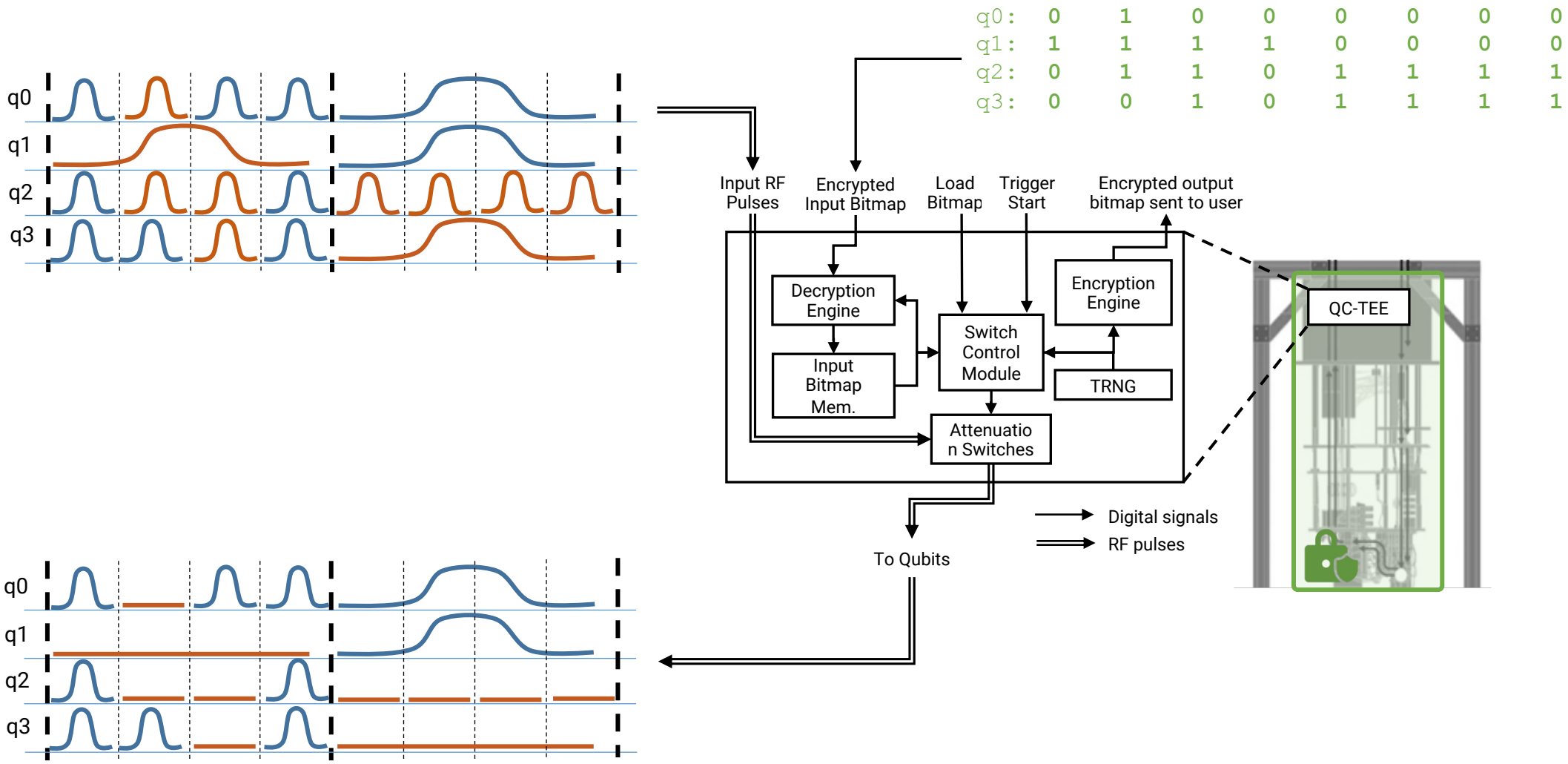
2. Circuit with decoy pulses:



3. Decoy pulses attenuated in the QPU



Trusted Hardware for Attenuation of Control Pulses



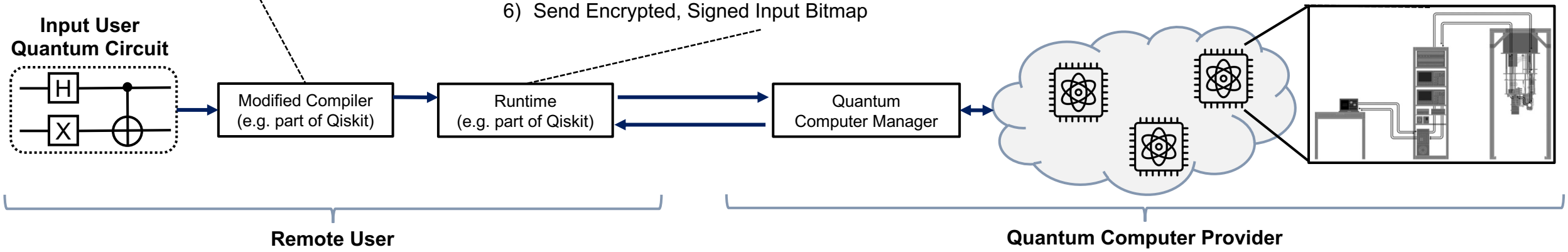
End-to-End Operation

- Software needs to generate the decoy pulses and the bitmap, while hardware attenuates the pulses in side the fridge:

1) Generate Control Pulses for User's Circuit
2) Mix Control Pulses with Decoy Pulses
3) Generate Input Bitmap

4) Encrypt Input Bitmap using AES-GCM and shared secret key which is generated using PQC-KEM and Public Key of the Backend
5) Send Transpiled Circuit including Mixed Decoy Pulses
6) Send Encrypted, Signed Input Bitmap

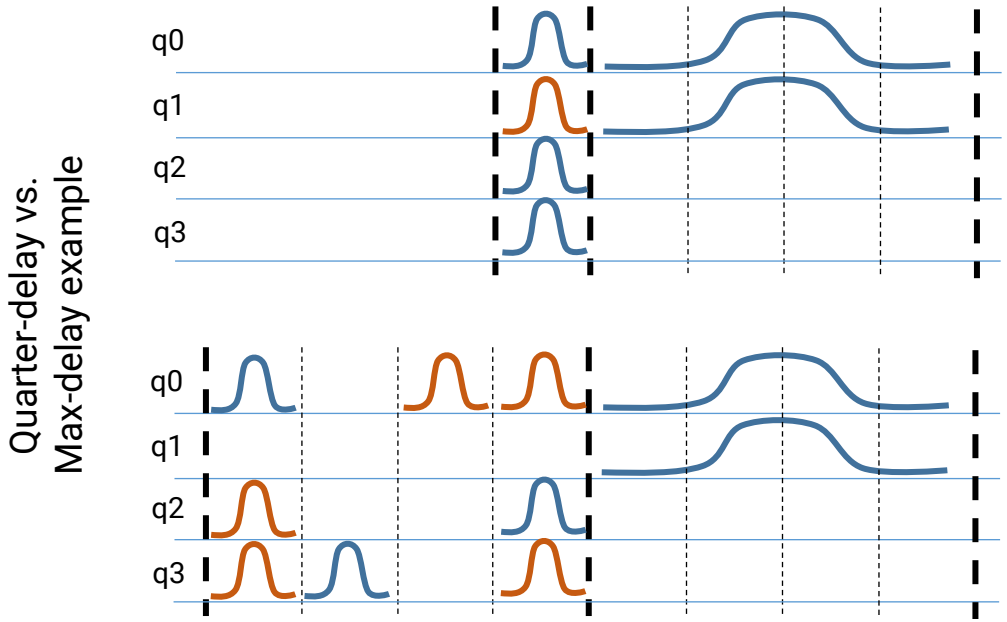
7) Decrypt Input Bitmap using AES-GCM and shared secret key generated in step 4
8) De-Mix Decoy Pulses
9) Execute Circuit



Evaluation Setup

- We evaluated different configurations:
  - Baseline
  - One-sixteenth-delay
  - Quarter-delay
  - Max-delay

- Further for all the levels, except baseline, we analyze results with and without a randomize-output option



- Evaluated on:
  - 7-qubit IBM Perth, to test small-scale benchmarks
  - 27-qubit IBM Algiers, through the Pay-As-You-Go plan on IBM Cloud to test medium-scale benchmarks
  - Aer simulator

Evaluation Using Variational Distance

- Worst case Variational Distance (VD) for the QASM-Bench benchmarks considering perfect RF switches and emulating imperfect RF switches (99.99% amplitude attenuation)

Obfuscation Level:	**max-delay**	
	w/o rand.-out.	**w/ rand.-out.**
Avg. VD (perfect RF switches)	0.2099	0.2453
Avg. VD (imperfect RF switches)	0.2652	0.2691

- Worst case Variational Distance (VD) for the QASM-Bench benchmarks for machines with different Quantum Volume (QV)

Obfuscation Level:	**max-delay**	
	w/o rand.-out.	**w/ rand.-out.**
Real IBM Perth ($QV = 32$)	0.2099	0.2453
Sim. IBM Perth ($QV = 32$)	0.1461	0.1481
Sim. IBM Mumbai ($QV = 64$)	0.1653	0.1705
Sim. IBM Cairo ($QV = 128$)	0.1286	0.1299

Evaluation of Benchmark Correctness

- Evaluation of correctness of 22 QASM-Bench benchmarks assuming imperfect RF switches (99.99% amplitude attenuation), all benchmarks show correct results

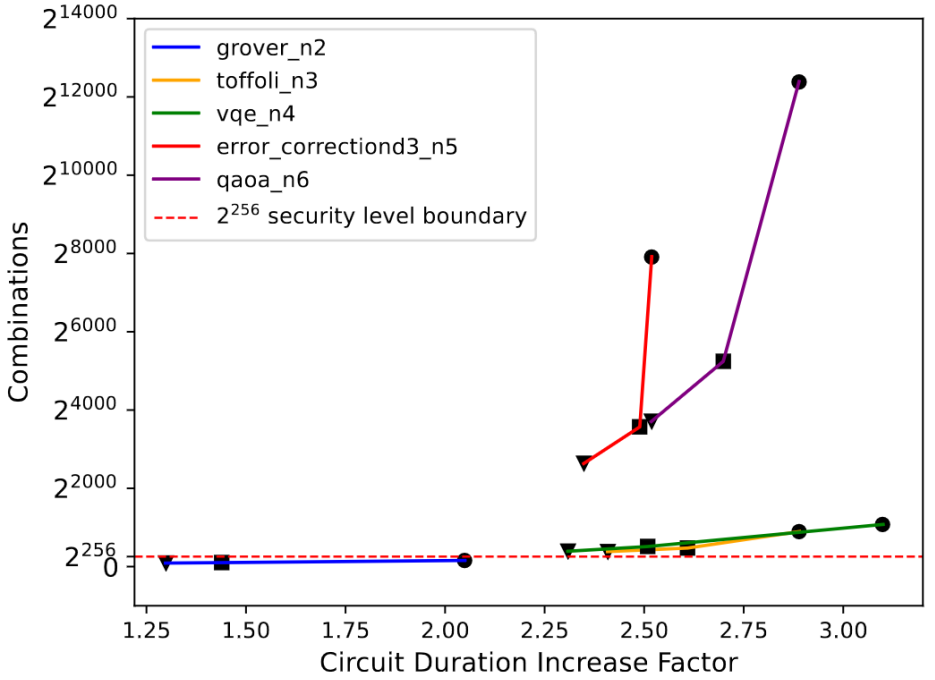
Benchmark	Qubits	Gates	CX	Depth	one-sixteenth-delay		quarter-delay		max-delay	
					w/o rand.-out.	w/ rand.-out.	w/o rand.-out.	w/ rand.-out.	w/o rand.-out.	w/ rand.-out.
deutsch	2	5	1	5	✓	✓	✓	✓	✓	✓
iswap	2	9	2	8	✓	✓	✓	✓	✓	✓
quantumwalks	2	11	3	8	✓	✓	✓	✓	✓	✓
grover	2	16	2	12	✓	✓	✓	✓	✓	✓
dnn	2	226	42	155	✓	✓	✓	✓	✓	✓
teleportation	3	8	2	7	✓	✓	✓	✓	✓	✓
qaoa	3	15	6	12	✓	✓	✓	✓	✓	✓
toffoli	3	18	6	13	✓	✓	✓	✓	✓	✓
linearsolver	3	19	4	12	✓	✓	✓	✓	✓	✓
fredkin	3	19	8	12	✓	✓	✓	✓	✓	✓
basis_change	3	53	10	22	✓	✓				
adder	4	23	10	12	✓	✓	✓	✓	✓	✓
bell	4	33	7	14	✓	✓	✓	✓	✓	✓
qft	4	36	12	9	✓	✓	✓	✓	✓	✓
variational	4	54	16	34	✓	✓				
vqe	4	89	9	28	✓	✓	✓	✓	✓	✓
basis_trotter	4	1626	582	815	✓	✓	✓	✓	✓	✓
qec_en	5	25	10	18	✓	✓	✓	✓	✓	✓
error_correctiond3	5	114	49	78	✓	✓	✓	✓	✓	✓
qaoa	6	270	54	110	✓	✓	✓	✓	✓	✓
bv	19	56	18	22	✓	✓	✓	✓	✓	✓
wstate	27	157	52	55	✓	✓	✓	✓	✓	✓

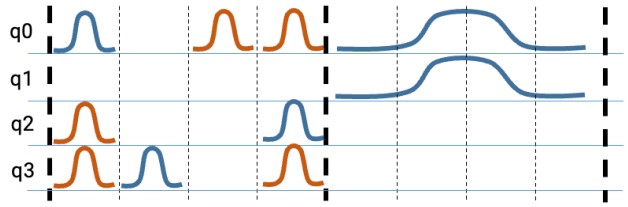
Security Analysis

- The cloud provider is able to see all the (real and decoy) control pulses, but needs to figure out which ones are real and which are decoy
- The complexity based on our analysis:

$$Comb = ((2^{n_{SubSlots}})^{n_{Slot_{SQ}}})^{n_{qubits}} \times ((2)^{n_{Slot_{CX}}})^{n_{SubCXInSlotCX}} \times ((2^{n_{SubSlotsInSlotCX}})^{n_{Slot_{CX}}})^{(n_{qubits}-2\times n_{SubCXInSlotCX})}$$

- Number of possibilities is much beyond 2^{256} combinations that cloud provider would have have to consider

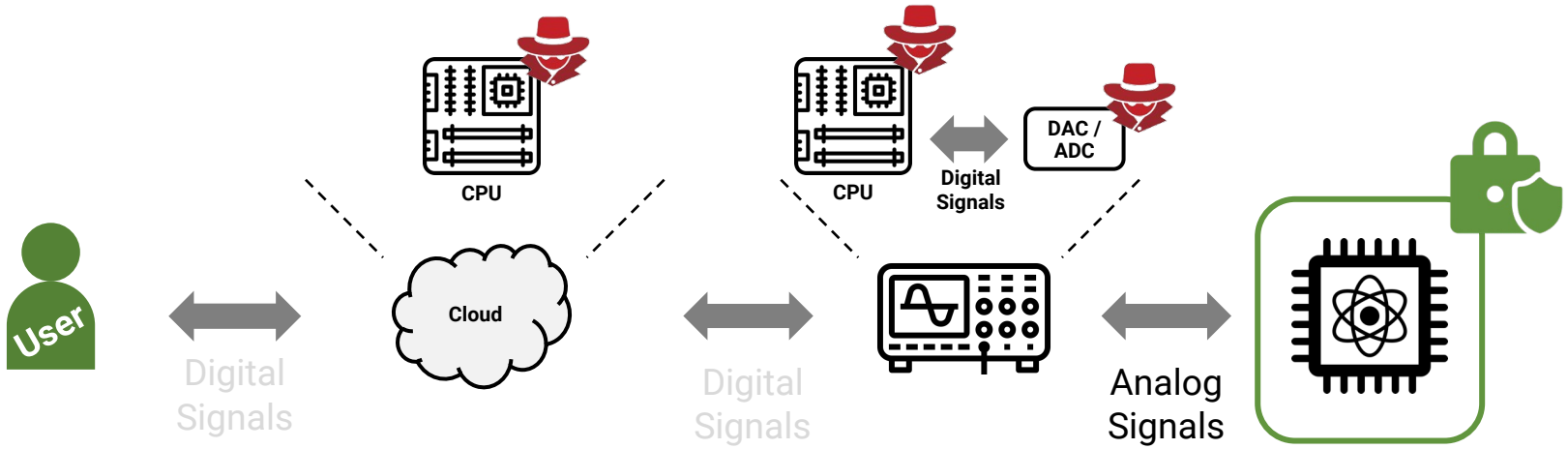




Trusted QPU with Dynamic Pulse Switching

- Maintain same threat model as QC-TEE
  - Trusted QPU
  - Components outside of QPU are not trusted

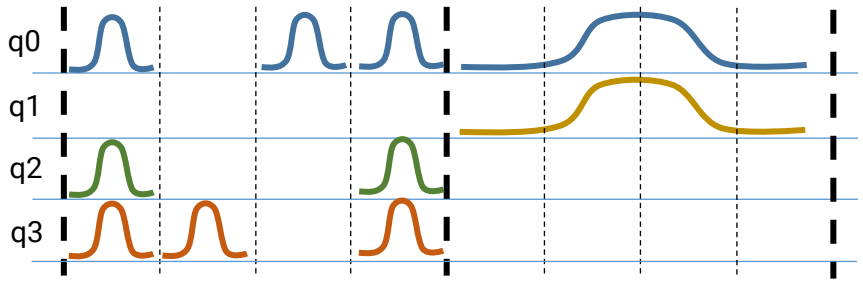
- **Key idea:** increase complexity (i.e. number of guesses attacker has to consider) by enabling switching of pulses between different channels



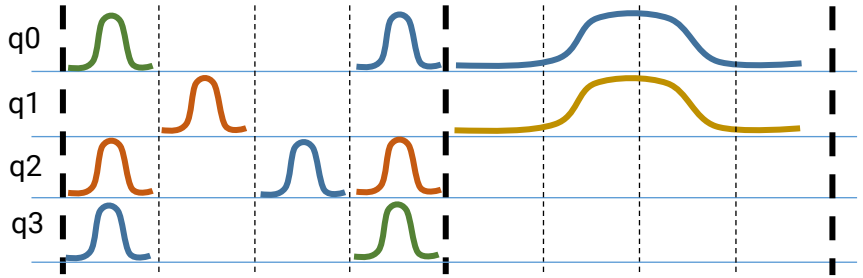
High-Level Idea of Switching Pulses

- We can increase confusion of the attacker by internally switching the control pulses

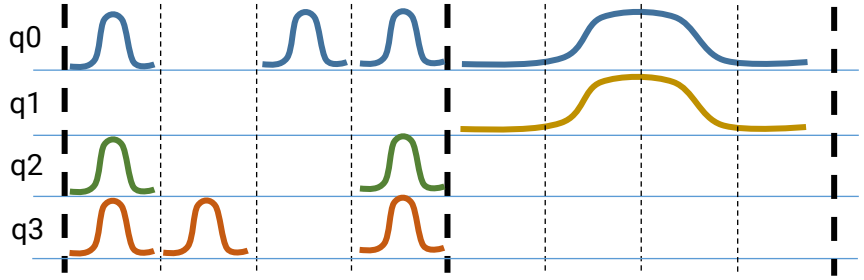
1. Input user circuit:



2. Circuit with pulses switched (as seen by cloud provider):

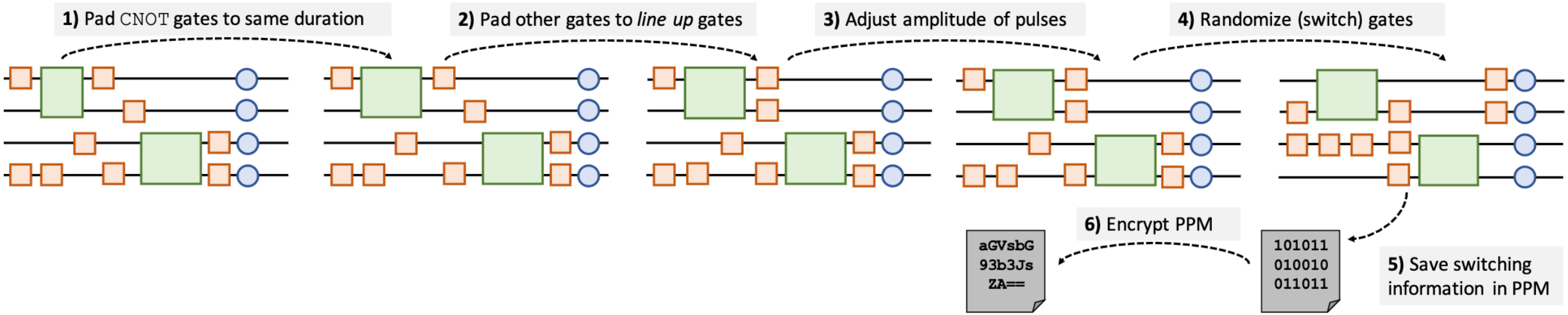


3. Pulses switched back inside the QPU:



CASQUE – Software Circuit Preparation

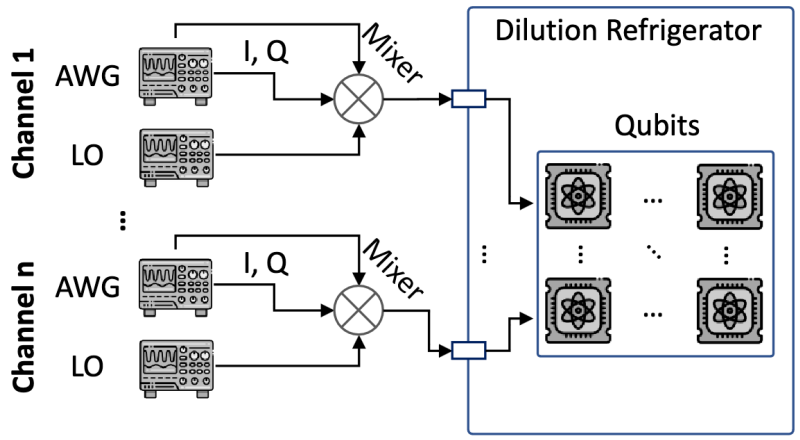
- On software side, quantum circuits need to be adjusted to enable the pulse switching, then pulses need to be randomly switched and the switching information encrypted for the QPU:



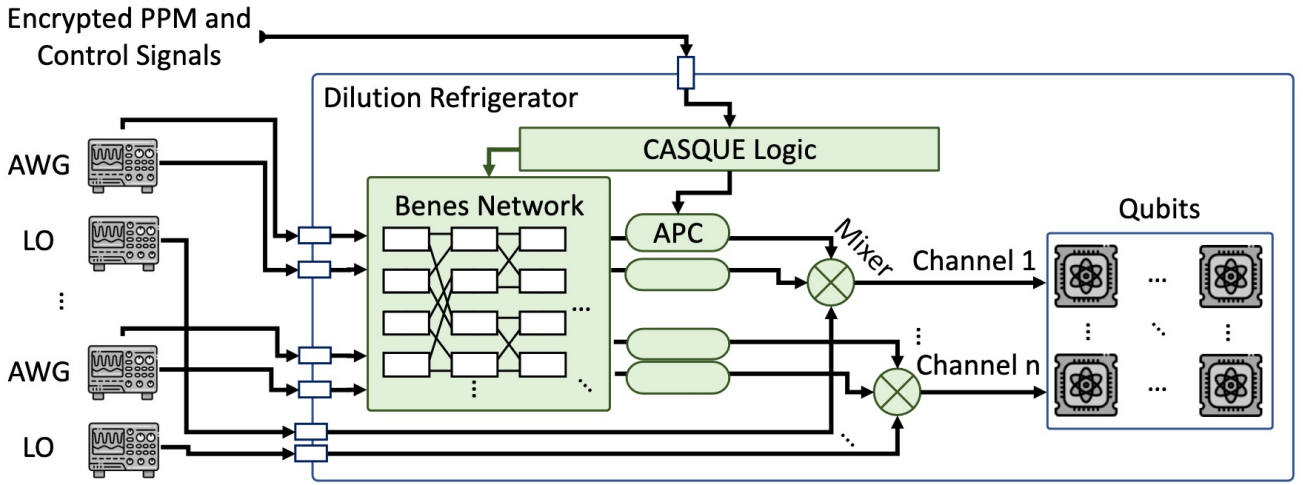
CASQUE – Proposed Hardware Changes

- Key challenge of switching the pulses between drive channels are the frequency, amplitude and phase
- CASQUE keeps signal generators outside the (trusted) fridge and the QPU, but adds new logic and mixers into the fridge

Typical QPU Setup



Proposed CASQUE Hardware Modifications



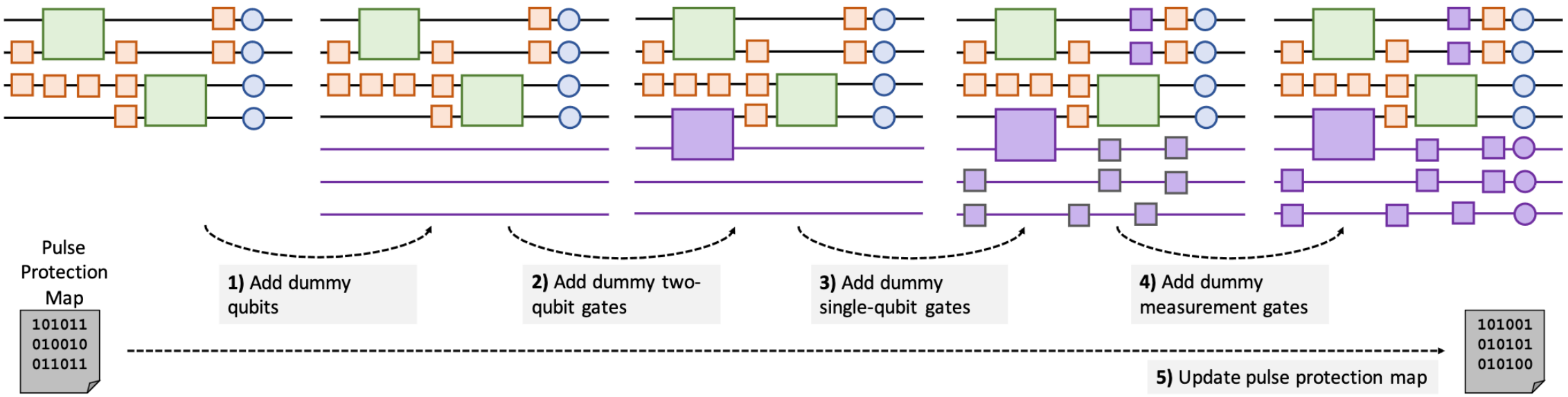
Evaluation Using Variational Distance

- Variational Distance (VD) for the QASM-Bench benchmarks considering perfect switching, decrease in fidelity is due to the extended circuit duration

Benchmark	Qubits	Gates	CNOT	Ref	VD
wstate	3	30	9	[27]	0.126
basis_change	3	53	10	[33]	0.116
variational	4	54	16	[33]	0.080
vqe	4	89	9	[27]	0.195
qec_en	5	25	10	[50]	0.642
error_correctiond3	5	114	49	[35]	0.184
simon	6	44	14	[4]	0.195
qaoa	6	270	54	[15]	0.203

CASQUE+ with Dummy Qubits

- CASQUE+ introduces idea of dummy qubits, any control pulses on the dummy qubits, or pulses switched to dummy qubits have no effect on the outcome of computation



Security Evaluation of CASQUE+

- Approximate attack complexity on selected QASM-Bench benchmark

Benchmark	Qubits	Gates	CNOT	Complexity			
				0 dummy qubits (CASQUE)	w/ 2 dummy qubits (CASQUE+)	w/ 4 dummy qubits (CASQUE+)	w/ 8 dummy qubits (CASQUE+)
wstate	3	30	9	2^{8}	2^{15}	2^{19}	2^{25}
basis_change	3	53	10	2^{43}	2^{66}	2^{80}	2^{100}
variational	4	54	16	2^{20}	2^{26}	2^{30}	2^{36}
vqe	4	89	9	2^{45}	2^{64}	2^{77}	2^{95}
qec_en	5	25	10	2^{17}	2^{22}	2^{26}	2^{32}
error_correctiond3	5	114	49	2^{103}	2^{132}	2^{155}	2^{191}
simon	6	44	14	2^{20}	2^{25}	2^{28}	2^{33}
qaoa	6	270	54	2^{268}	2^{324}	2^{365}	2^{428}

Other Possible Secure Quantum Computer Architectures

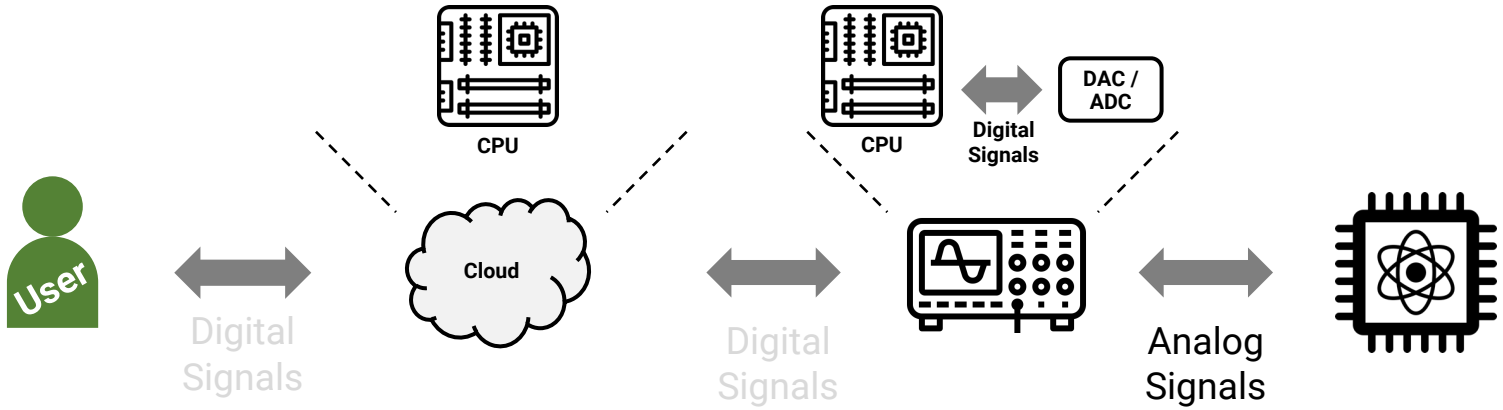
Blind Quantum Computation

QC-TEE & CASQUE

New Solution: Trusted Controller

Cloud (CPU)	Controller (CPU)	Controller (DAC)	QPU
Untrusted	Untrusted	Untrusted	Untrusted
Untrusted	Untrusted	Untrusted	Trusted
Untrusted	Untrusted	Trusted	Untrusted
Untrusted	Untrusted	Trusted	Trusted
Untrusted	Trusted	Untrusted	Untrusted
Untrusted	Trusted	Untrusted	Trusted
Untrusted	Trusted	Trusted	Untrusted
Untrusted	Trusted	Trusted	Trusted

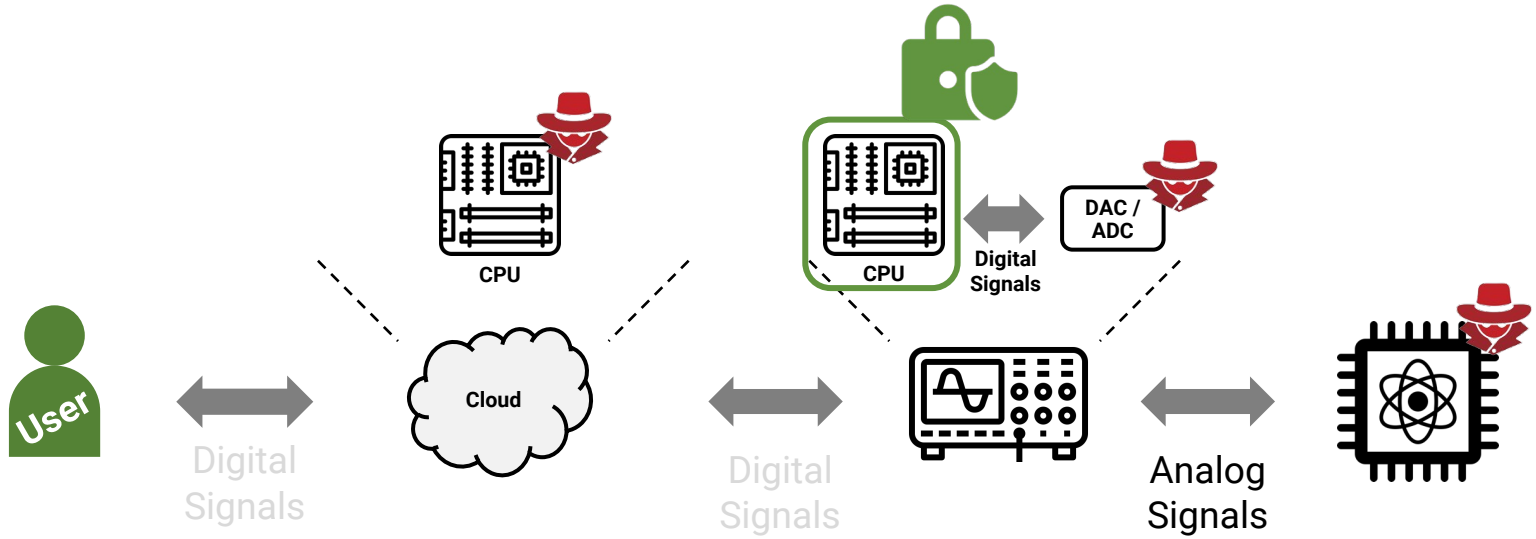
Cloud (CPU)	Controller (CPU)	Controller (DAC)	QPU
Trusted	Untrusted	Untrusted	Untrusted
Trusted	Untrusted	Untrusted	Trusted
Trusted	Untrusted	Trusted	Untrusted
Trusted	Untrusted	Trusted	Trusted
Trusted	Trusted	Untrusted	Untrusted
Trusted	Trusted	Untrusted	Trusted
Trusted	Trusted	Trusted	Untrusted
Trusted	Trusted	Trusted	Trusted



Quantum Program Obfuscation with Trusted Controller

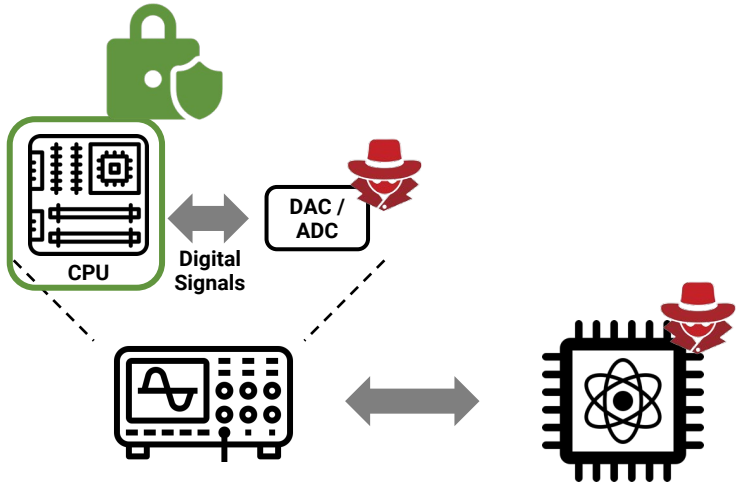
- Demonstrate how trusted controller could provide protections from untrusted cloud provider or malicious insiders

- Leverage existing classical security solutions such as Intel SGX
  - Avoid modifications to QPU
  - Avoid assumptions about trusted QPU
  - Use well-studied Intel SGX

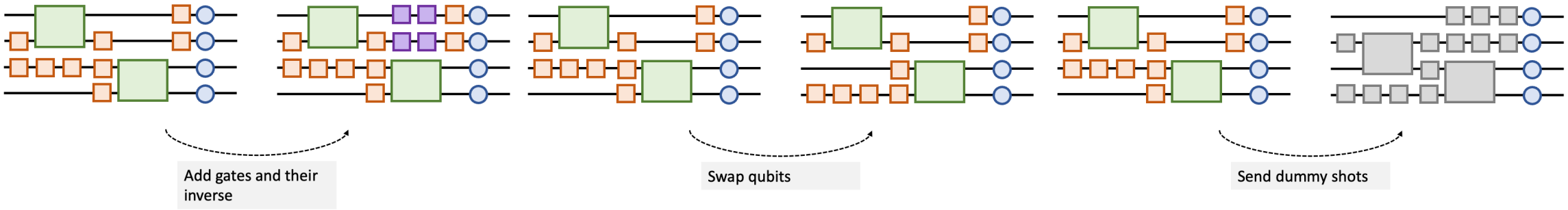


Trusted Controller's Possible Operations

- Adversary (cloud provider) can see all the control pulses leaving the (trusted) controller, limiting what the controller can do

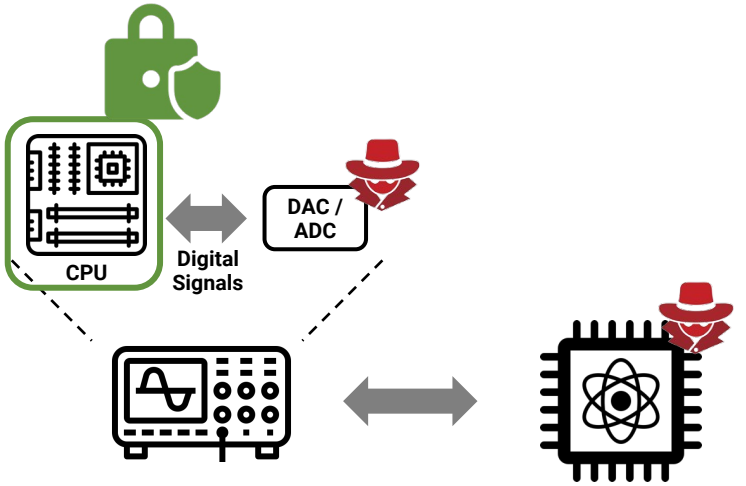


- Possible operations done by the trusted controller:

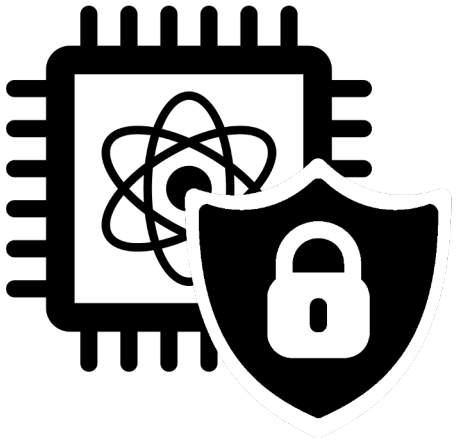


Informal Security Analysis

- Cloud provider always sees the output and has total knowledge of the pulses going to the QPU
- Informally, at best $O(n)$ complexity for attacker to guess the program, n is number of shots



- Still some benefits
  - Increase n by mixing shots from different users
  - Send different shots to different QPUs
  - Future ideas, leverage metamorphic properties of programs to further hide the true computation
- … and leverage existing classical security solutions such as Intel SGX
  - Avoid modifications to QPU
  - Avoid assumptions about trusted QPU
  - Use well-studied Intel SGX



Tutorial on Security of Quantum Computing Systems

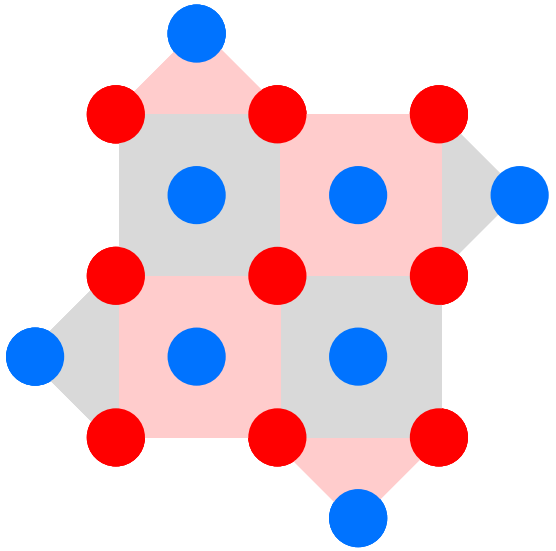
Fault-Tolerant Quantum Computing (FTQC) Security

Computer Architecture and Security Lab (CASLAB)

Northwestern University

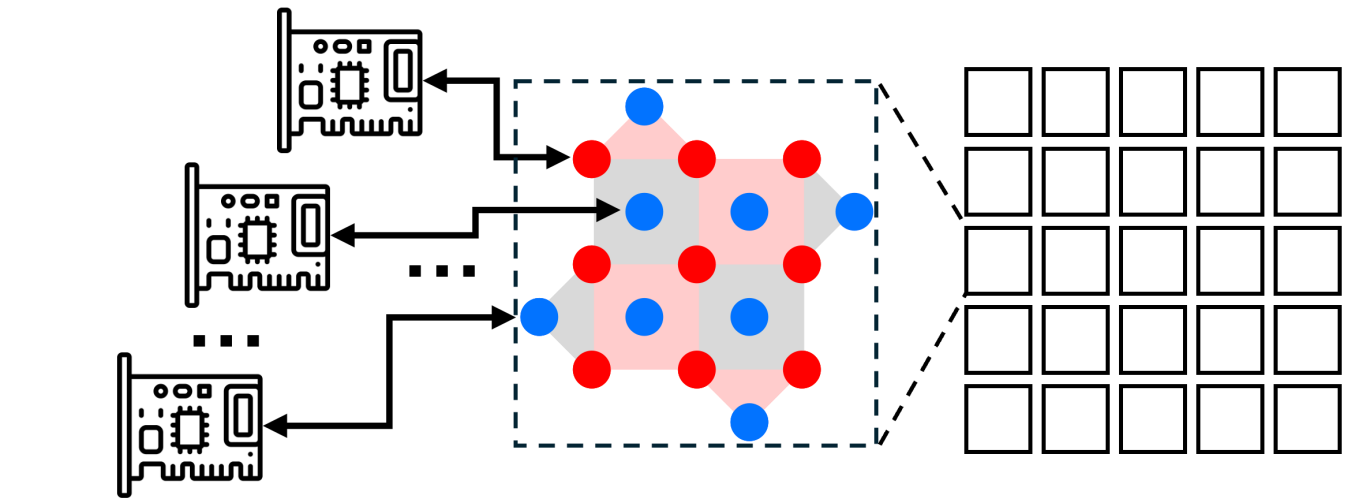
Fault-Tolerant Quantum Computing

- Fault-Tolerant Quantum Computing (FTQC) develops error correction and mitigation techniques to limit noise and errors in quantum computers
- FTQC produces logical qubits with reduced error rates by using additional physical qubits to produce redundancies to faults
- Many error correction techniques exist, one very promising is surface code, that uses data and stabilizer qubits:
  - *Data qubits*: hold the actual quantum information (the logical quantum states)
  - Ancilla (or stabilizer) qubits: used to measure error syndromes without disturbing the data qubits



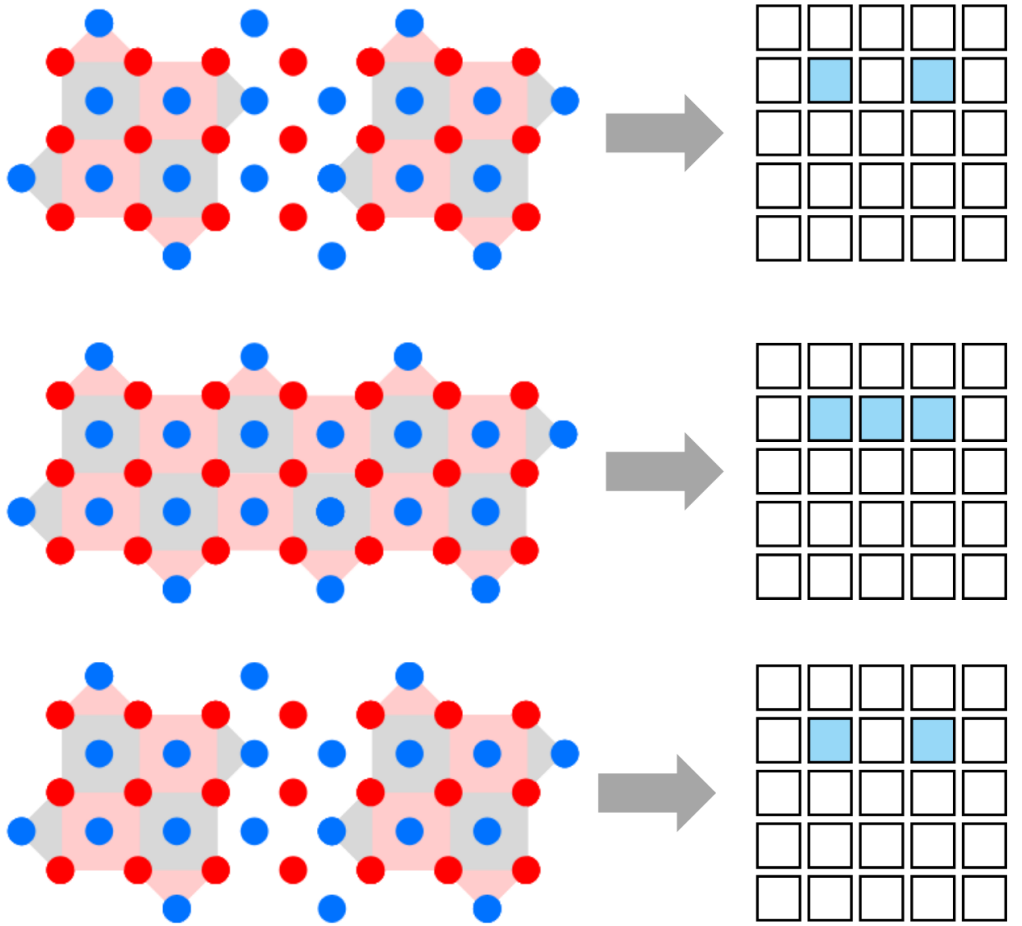
FTQC Control Infrastructure

- Controllers, containing Quantum Error Correction decoders and other logic, interface to the physical qubits, each group of physical qubits makes up a patch or a logical qubit



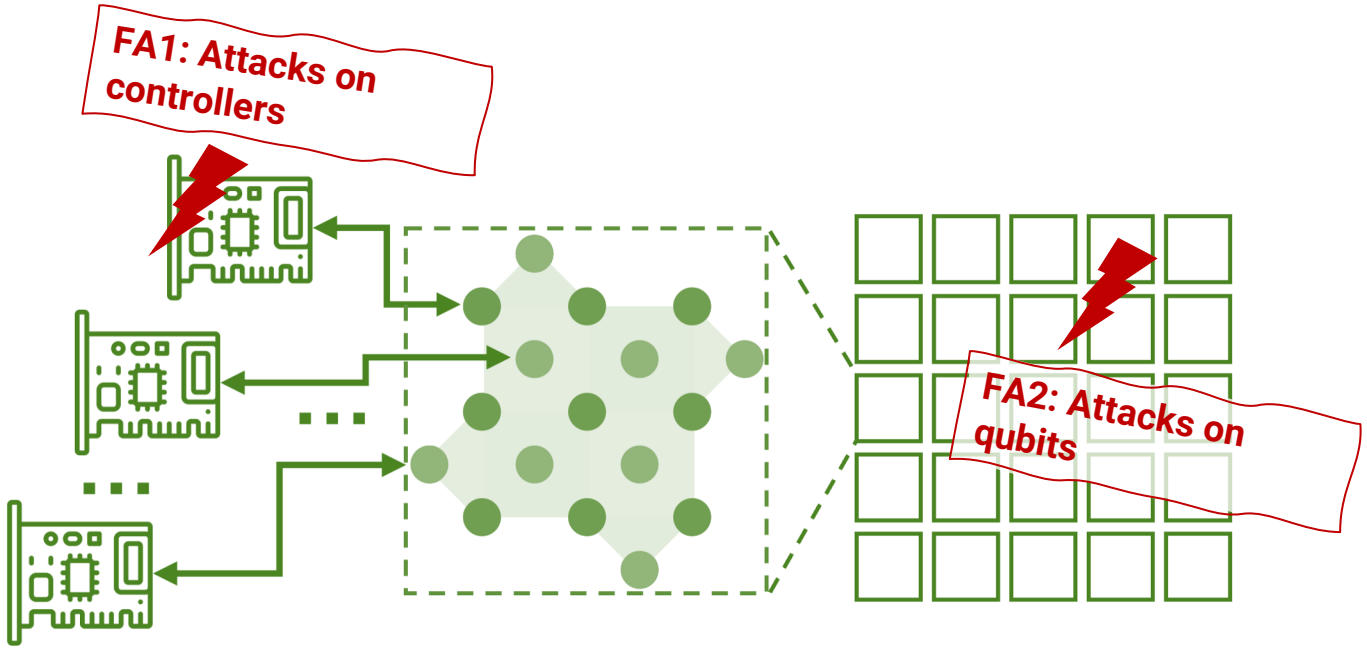
Surface Code Lattice Surgery

- Executing two-qubit gates requires merging and splitting the regions that are used to hold the individual logical qubits
    - Start with two logical qubits
    - Patches are merged via lattice surgery techniques
    - Split after two-qubit operation is performed



FTQC Security Threats

- Security threats to fault-tolerant quantum computers



FTQC Security

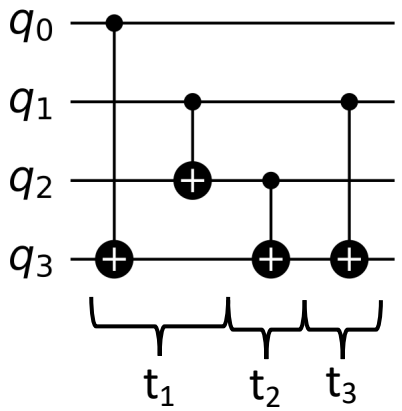
- FTQC quantum computers are still “computers” making them vulnerable to various security attacks, but:

- **Pros of FTQC quantum computers:**
  - Error correction should protect from noise or errors
    - Limited impact of fault injection attacks among logical qubits
    - Limited impact of side channels as user only access logical (error-corrected) qubits

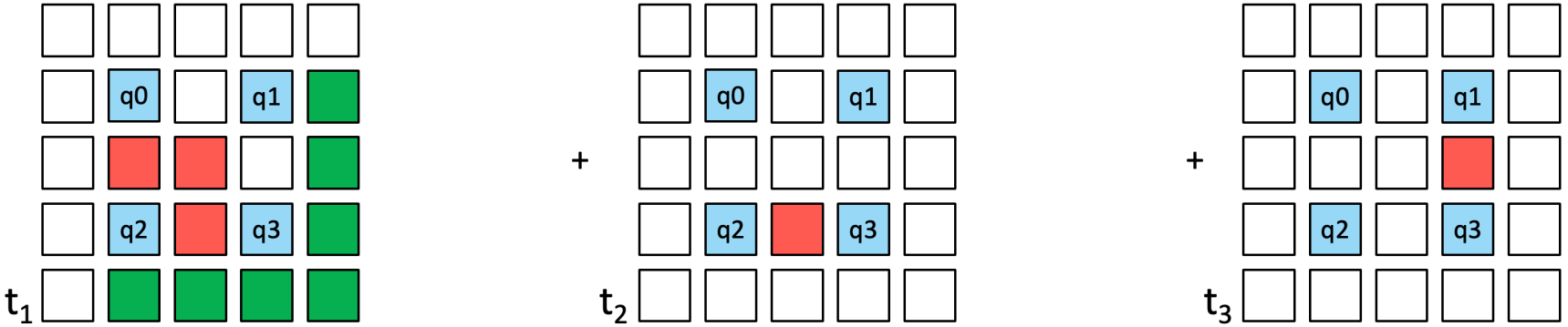
- **Cons of FTQC quantum computers:**
  - Still rely on controllers to operate qubits
    - Fault injection can affect error correction or the gate operations
    - Side channels can leak information about gate operations
  - Possible increased vulnerability due to many physical qubits
    - More qubits to attack or target
    - Tracing attacks lattice surgery – two qubit operations
    - Timing attacks on magic states – single qubit operations

Tracing Attacks on FTQC and Lattice Surgery

- Tracing merging and splitting operations can reveal two qubit operations
- Quantum circuit:



- Activity on quantum chip:

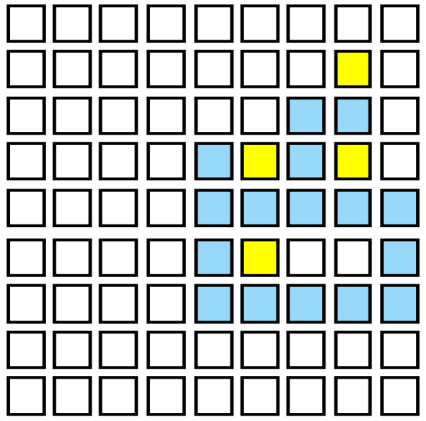


Reference:
Trochatos , et al. (2025)

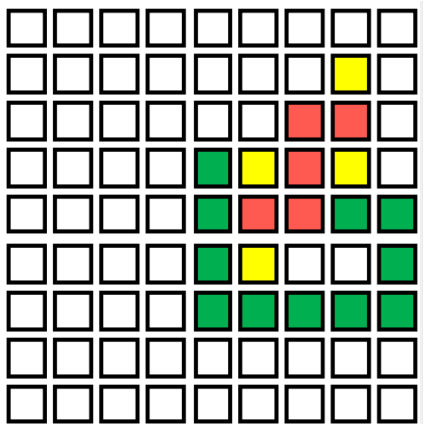
Tracing FTQC Chip Activity

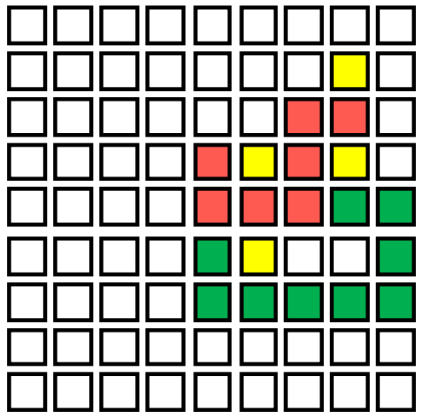
- There can ambiguity in the traces

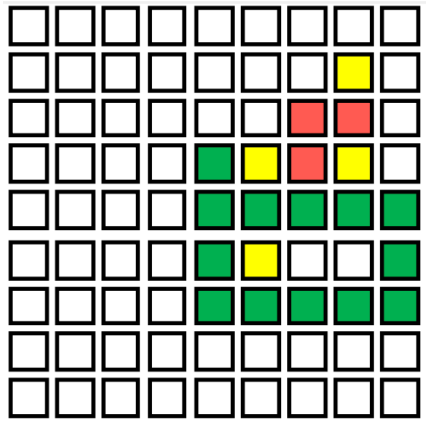
Activity observation:



Possible traces:



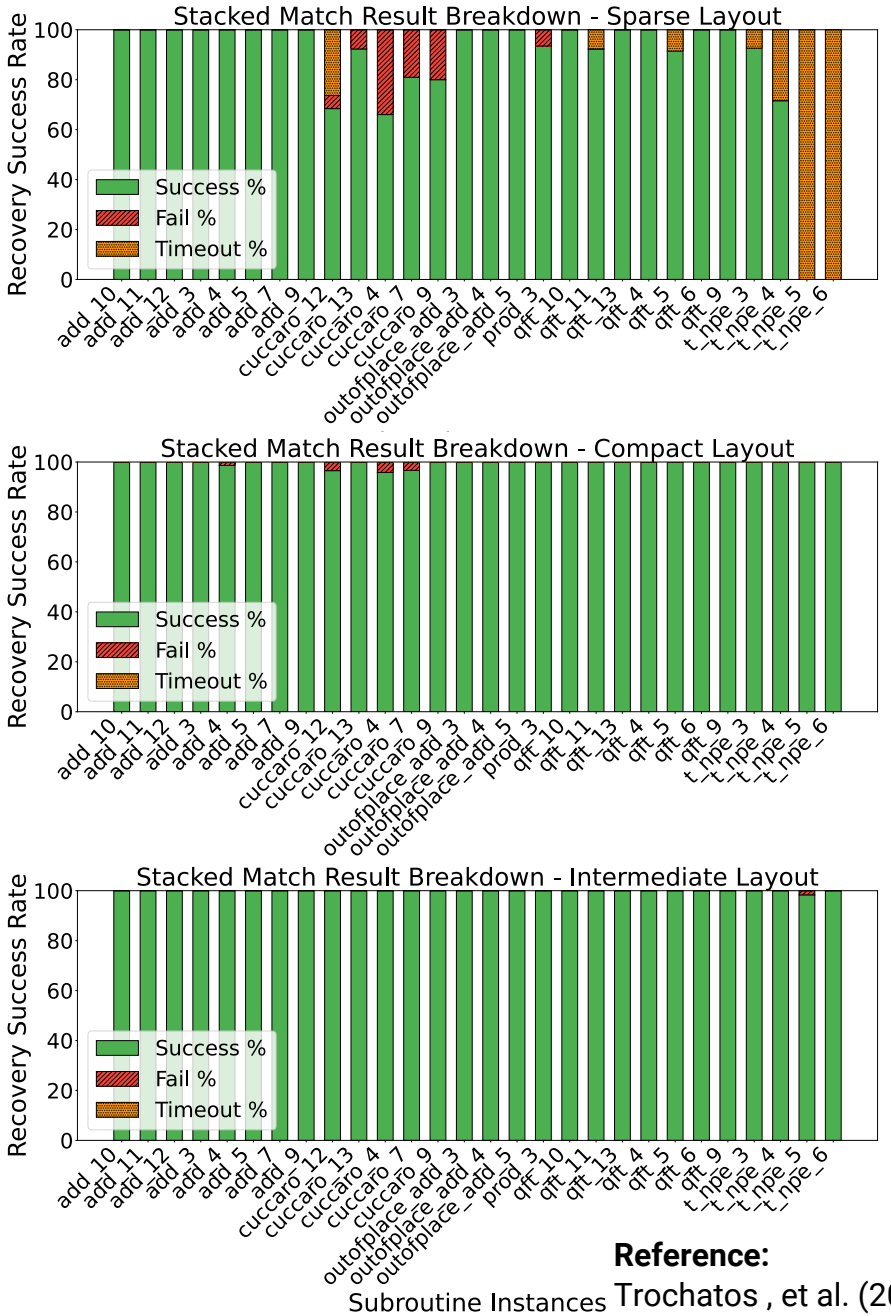




Reference:
Trochatos , et al. (2025)

FTQC Tracing Attack

- Tracing FTQC activity can reveal connections between qubits as they change over time
  - Each merge and split is a two-qubit operation
  - Can build Directed Acyclic Graph (DAG) of the circuit from the traces
  - DAGs can be used as signatures of the algorithms or functions
  - Knowing a databases of common functions, these can be identified in the DAGs

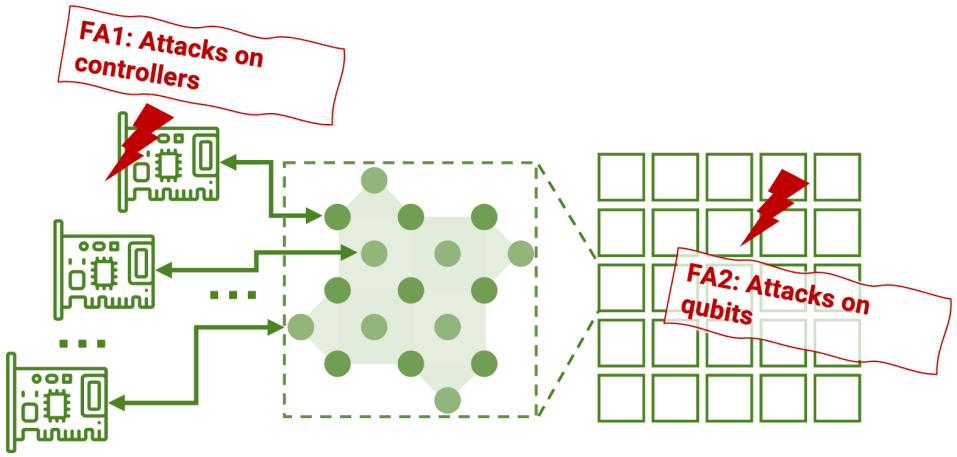


Reference:
Trochatos , et al. (2025)

FTQC Attacks Summary and Outlook

Some attacks have been demonstrated:

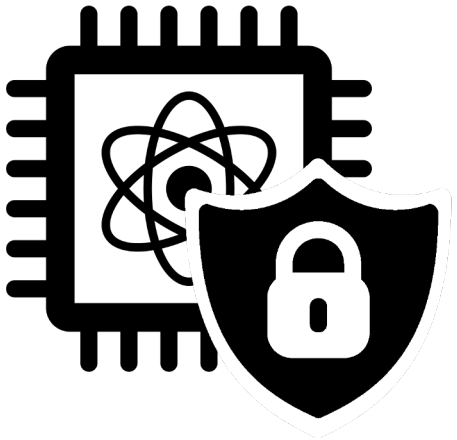
- Trace-based side-channels (2025)
- Attacks on Reinforcement Learning based decoders (2025)



No defenses yet have been evaluated

Future directions in this research:

- Extend both side channel and fault injection attack analysis
- Prototyping defenses



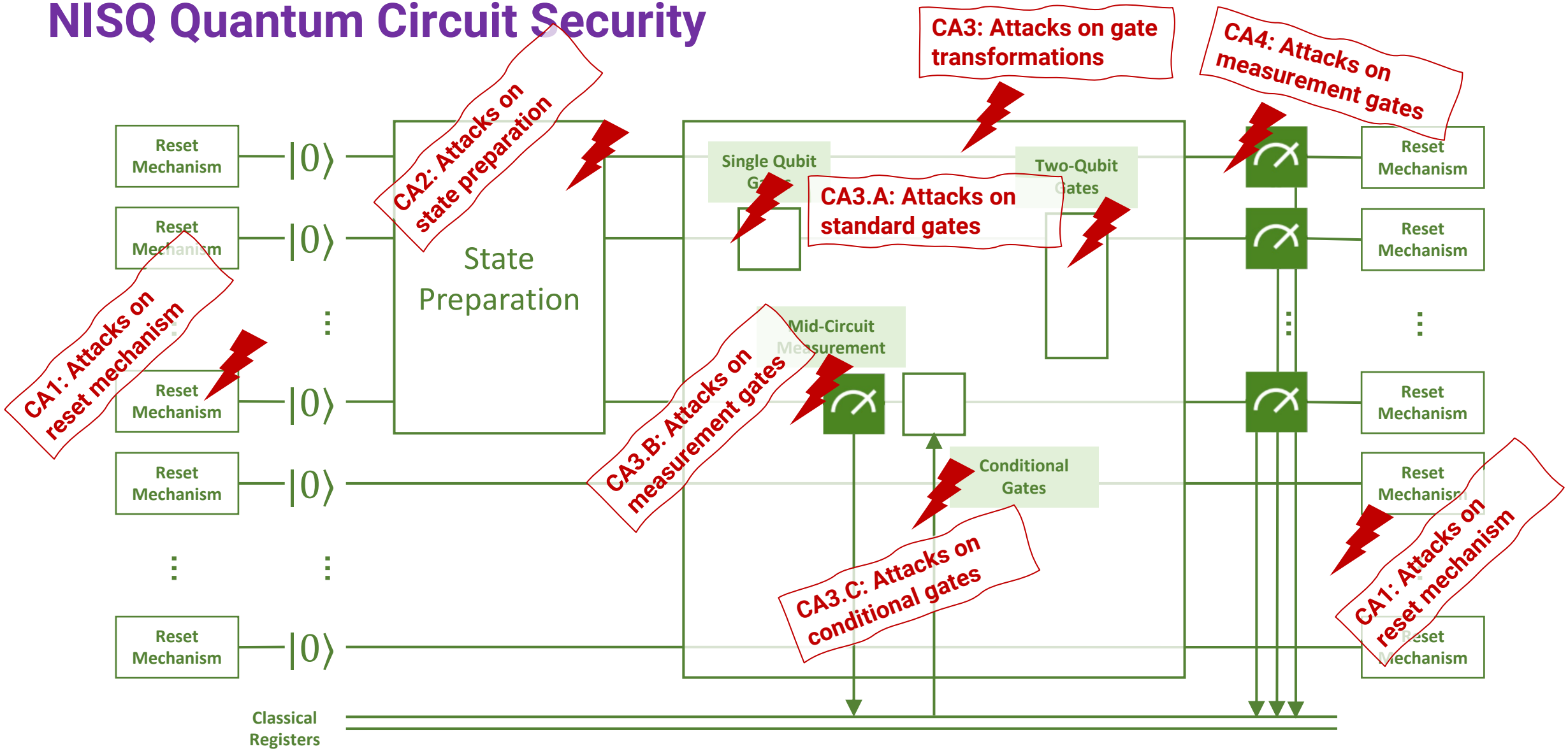
Tutorial on Security of Quantum Computing Systems

Summary and Conclusion

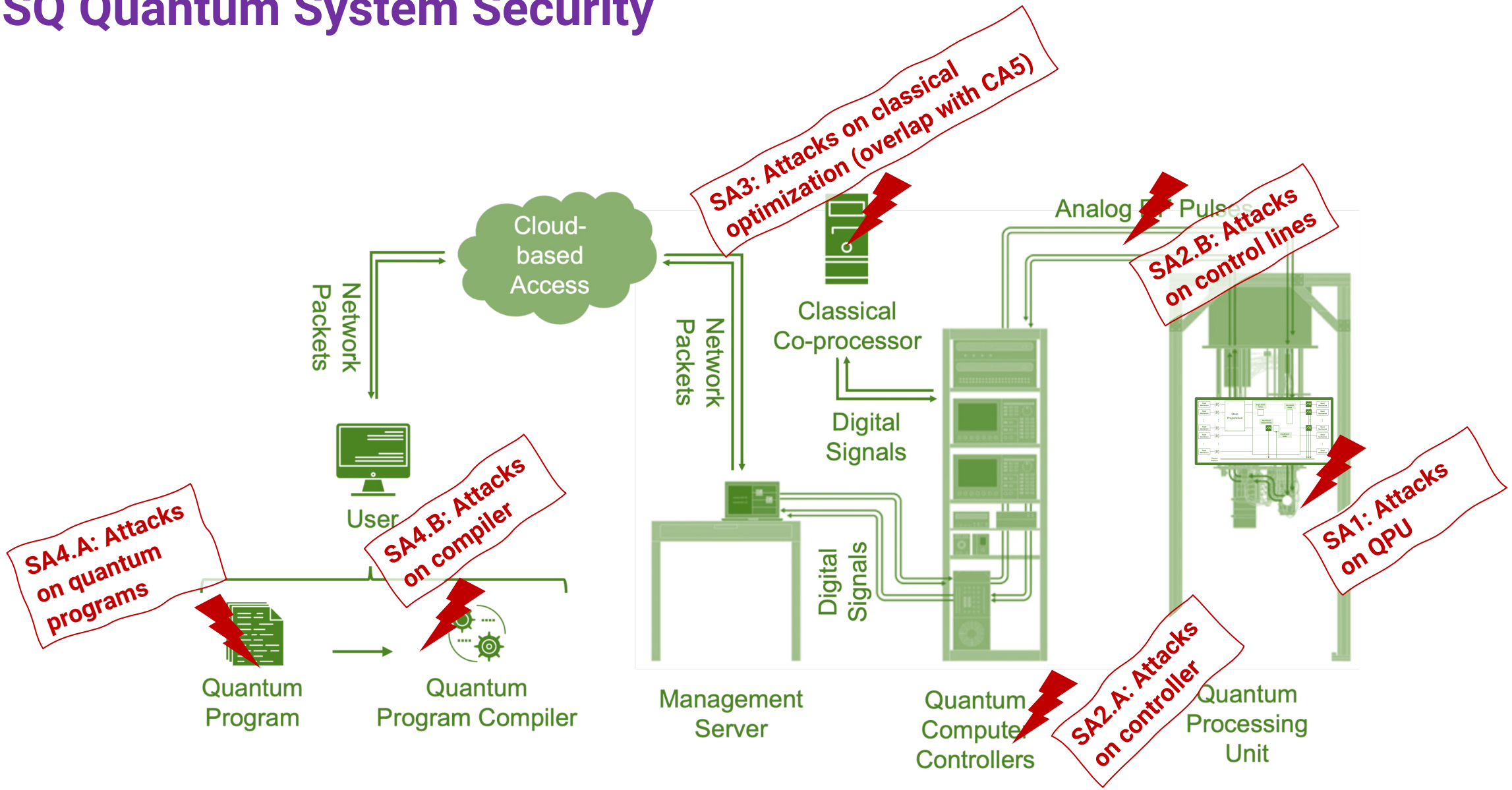
Computer Architecture and Security Lab (CASLAB)

Northwestern University

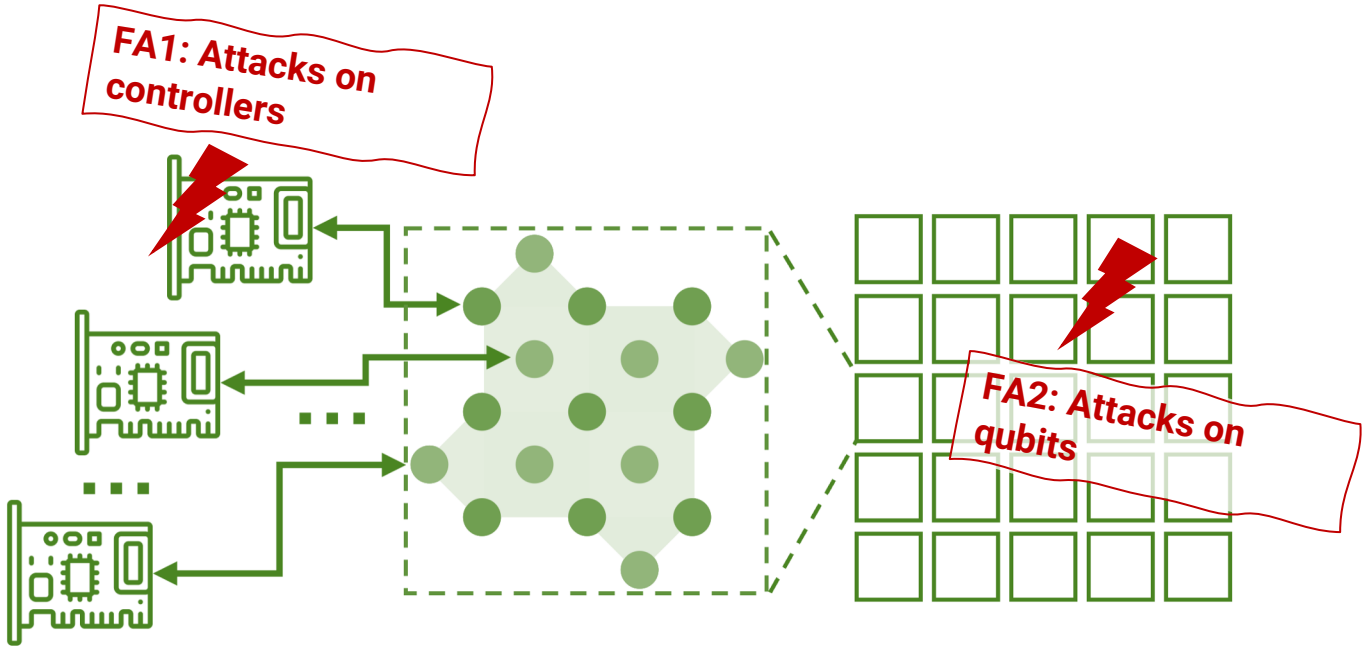
NISQ Quantum Circuit Security



NISQ Quantum System Security

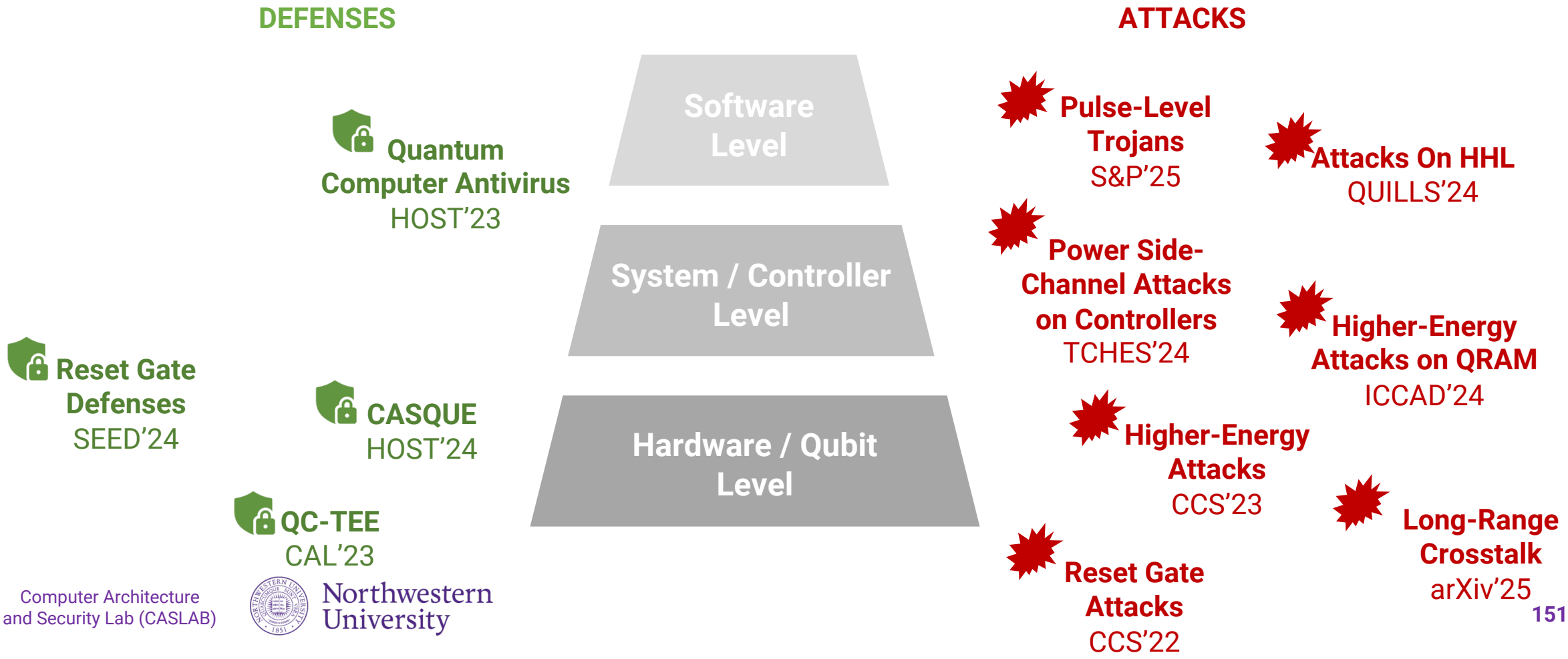


FTQC Security



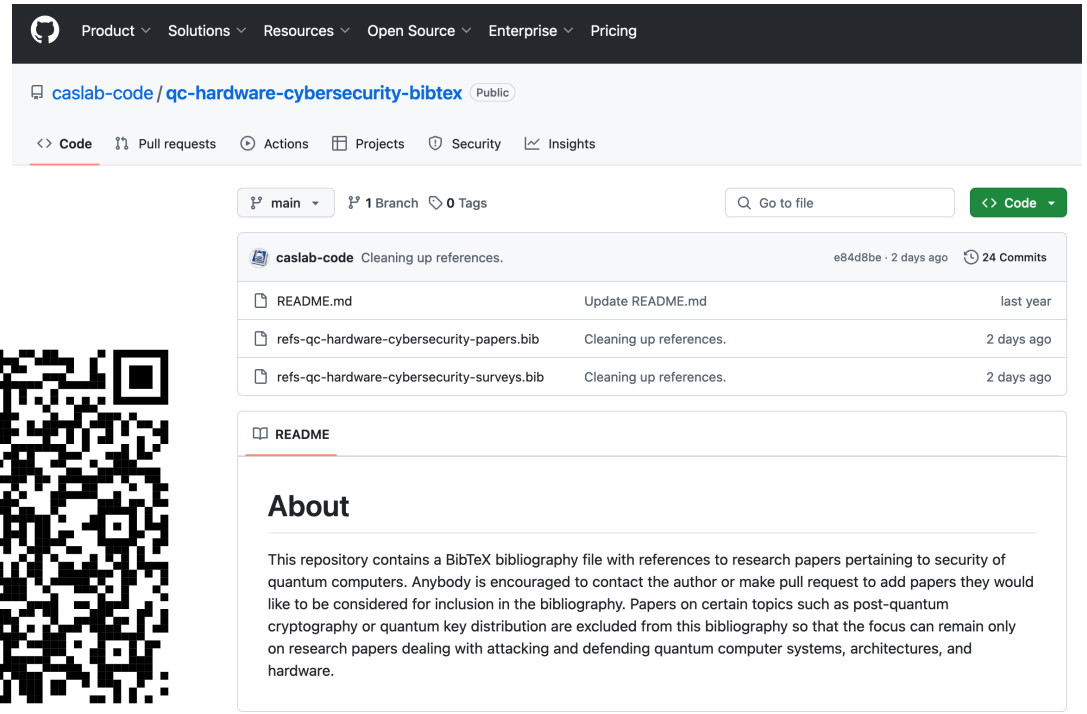
Many Attacks and Defenses Exist, But Many Questions Remain

Secure Quantum Computing is an emerging field that aims to address security and privacy concerns arising from the delegation of quantum programs to untrusted remote servers.



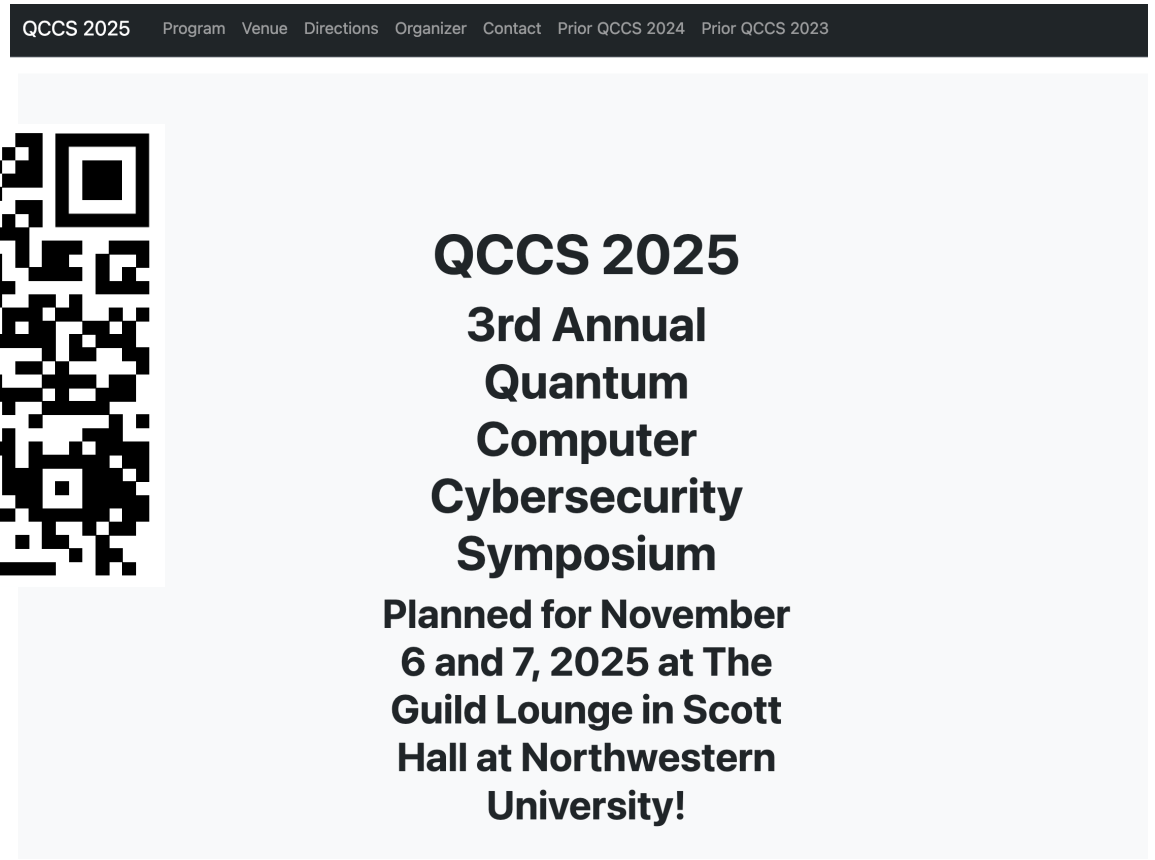
Quantum Computer Cybersecurity Resources

- Tutorial resources:
  - **Quantum Computer Hardware Cybersecurity BibTeX**
    - https://github.com/caslab-code/qc-hardware-cybersecurity-bibtex
    - A bibliography file containing references to Quantum Computer Hardware Cybersecurity research papers
    - Papers from various research groups, superset of work presented at this tutorial
    - Regularly updated



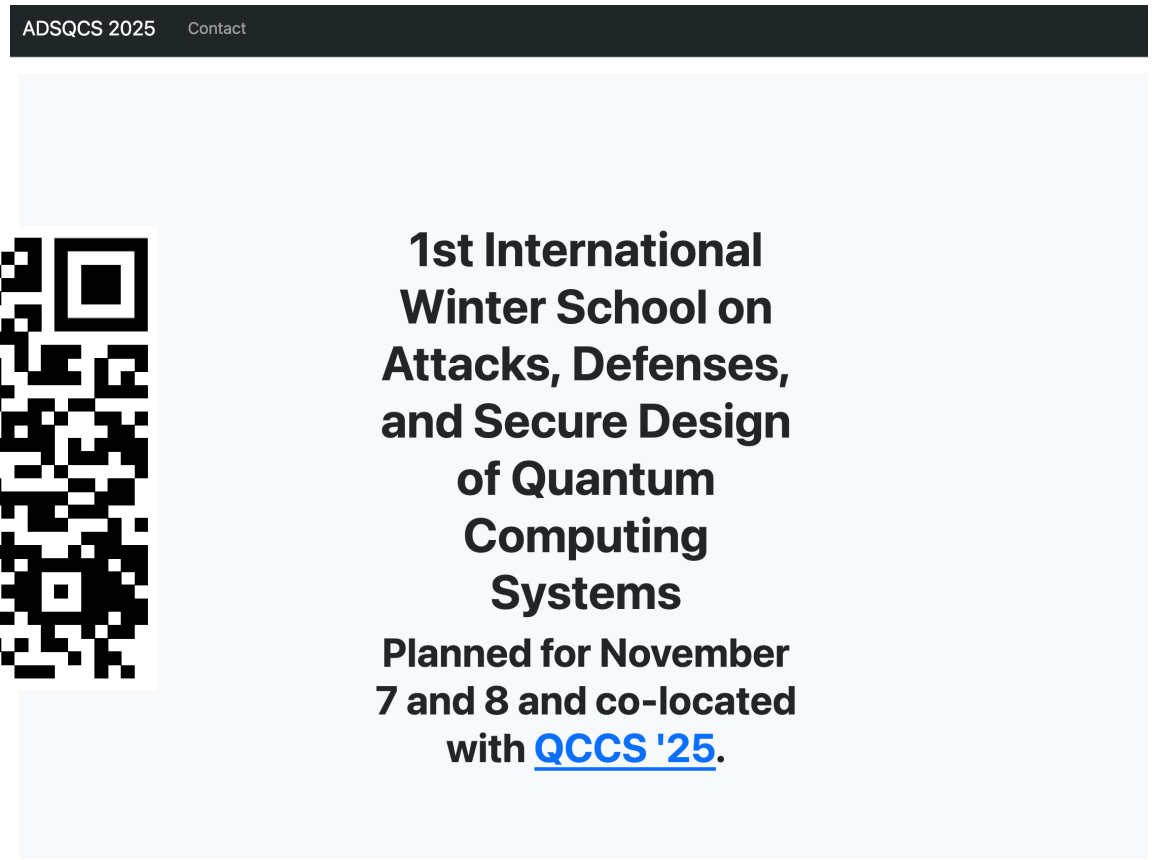
Quantum Computer Cybersecurity Resources

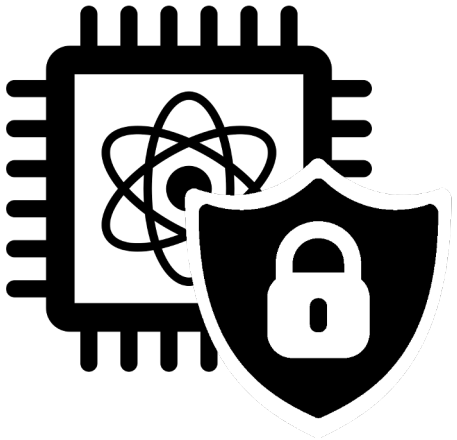
- Quantum Computer Cybersecurity Symposium (QCCS ‘25)
- Web: https://caslab.io/events/qccs



Quantum Computer Cybersecurity Resources

- 1st International Winter School on Attacks, Defenses, and Secure Design of Quantum Computing Systems (ADSDQC ‘25)
- Web: https://caslab.io/events/adsqcs
- In-person, hands-on instructions and experimentation!





Thank You!

Quantum Week Tutorial

August 2025

Prof. Jakub Szefer – jakub.szefer@northwestern.edu

Electrical and Computer Engineering

Northwestern University

Computer Architecture and Security Lab (CASLAB)

Northwestern University