Quantum PUF for Security and Trust in Quantum Computing

Koustubh Phalak, Abdullah Ash- Saki, *Graduate Student Member, IEEE*, Mahabubul Alam, *Student Member, IEEE*, Rasit Onur Topaloglu, *Senior Member, IEEE*, and Swaroop Ghosh, *Senior Member, IEEE*

***Abstract*— Quantum computing is a promising paradigm to solve computationally intractable problems. Various companies such as, IBM, Rigetti and D-Wave offer quantum computers using a cloud-based platform that possess several interesting features namely, (i) quantum hardware with various number of qubits and coupling maps exist at the cloud end that offer different computing capabilities; (ii) multiple hardware with identical coupling maps exist in the suite; (iii) coupling map of larger hardware with more number of qubits can fit the coupling map of many smaller hardware; (iv) the quality of each of the hardware is distinct; (v) user cannot validate the origination of the result obtained from a quantum hardware. In other words, the user relies on the scheduler of the cloud provider to allocate the requested hardware; (vi) the queue of quantum programs at the cloud end is typically long and maximizing the throughput, which is the key to reducing costs and helping the scientific community in their explorations. The above factors motivate a new threat model with following possibilities: (a) in future, less-trustworthy quantum computers from 3rd parties can allocate poor quality hardware to save on cost or towards satisfying their falsely-advertised qubit or quantum hardware specifications; (b) the workload scheduling algorithm could have a bug or malicious code segment which will try to maximize throughput at the cost of allocation to poor fidelity hardware. Such bugs are possible for trustworthy providers; (c) a rogue employee in trusted cloud vendor could try to sabotage the vendor's reputation by degrading the user compute fidelity just by tampering with the scheduling algorithm or rerouting the program; (d) a rogue employee can steal information by redirecting the programs to a 3rd party quantum hardware where they have full control. If the allocated hardware is inferior in quality, the user will suffer from poor quality result or longer convergence time. We propose two flavors of a Quantum Physically Unclonable Function (QuPUF) to address this issue- one based on superposition and another based on decoherence. Our experiments on real quantum hardware reveal that temporal variations in qubit quality can degrade the quality of the proposed QuPUF. We add a parametric rotation to the QuPUF for stability. Experiments on real IBM quantum hardware show that the proposed QuPUF can achieve inter-die Hamming Distance (HD) of 55% and intra-HD as low as 4%, as compared to ideal cases of 50% and 0% respectively. The proposed QuPUFs can also be used as a standalone solution for any other application.**

***Index Terms*— Quantum computing, security, quantum PUF.**

I. Introduction

QUANTUM computing can solve computationally intractable problems in domains e.g., finance, traffic flow and power grid by exploiting superposition and entanglement properties. Various qubit technologies are being explored including superconducting, Ion Trap and single electron by academia and industry to develop scalable quantum computers. While the best scalable quantum technology is an active area of research, design community is exploring the quantum computers offered by various companies such as, IBM, Rigetti and D-Wave to solve optimization problems. Currently, the access to quantum computers is provided through cloud-based platform where a suite of quantum computers are available for the users to solve their problems. The users can compile their circuits for a particular hardware and transmit to the cloud which enters a queue. The scheduling algorithm allocates the programs from the queue using a pre-defined allocation policies such as, fair share allocation [1]. Once the experiment is concluded, the results are sent back to the user. Since the noisy computers are less powerful and limited in the number of qubits, various hybrid algorithms such as, Quantum Approximate Optimization Algorithm (QAOA) and Variational Quantum Eigensolver (VQE) are pursued where a classical computer drives the parameters of a quantum algorithm/circuit iteratively. The goal of the classical computer is to find the right set of parameters that can drive the quantum algorithm towards the optimal solution for a given problem. For high-quality hardware with reliable qubits, the algorithm is expected to converge faster i.e., with less number of iterations. However, the quantum computers with more number of qubits and/or high quality qubits typically come at a higher cost. Therefore, securing the hardware with desired quality is very important to solve a certain problem within a desired deadline. However, the user gets little/no

Manuscript received December 8, 2020; revised February 22, 2021 and April 27, 2021; accepted April 28, 2021. Date of publication May 3, 2021; date of current version June 14, 2021. This work was supported in part by the National Science Foundation (NSF) under Grant CNS-1722557, Grant CCF-1718474, Grant OIA-2040667, Grant DGE-1723687, and Grant DGE-1821766; in part by the Penn State Institute for Computational and Data Sciences; and in part by the Penn State Huck Institute of the Life Sciences. This article was recommended by Guest Editor Kanad Basu. *(Corresponding author: Koustubh Phalak.)*

Koustubh Phalak, Abdullah Ash- Saki, Mahabubul Alam, and Swaroop Ghosh are with the School of Electrical Engineering and Computer Science (EECS), Pennsylvania State University, University Park, PA 16802 USA (e-mail: krp5448@psu.edu).

Rasit Onur Topaloglu is with IBM, Yorktown Heights, NY 12533 USA.

Color versions of one or more figures in this article are available at https://doi.org/10.1109/JETCAS.2021.3077024.

Digital Object Identifier 10.1109/JETCAS.2021.3077024

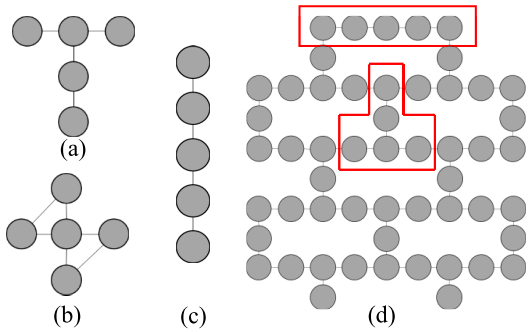


Fig. 1. The structures of various quantum computers, (a) ibmq_london, ibmq_burlington, ibmq_essex, ibmq_vigo, etc.; (b) ibmq_yorktown; (c) ibmq_santiago; and (d) ibmq_rochester. It can be noted that rochester consists of several isomorphic graphs of ibmq_london, ibmq_burlington, ibmq_santiago etc. Therefore, ibmq_rochester hardware can accommodate multiple workloads meant for ibmq_london, ibmq_santiago etc. in parallel.

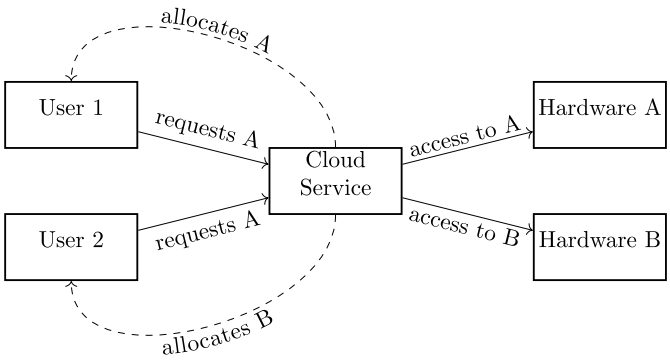


Fig. 2. Conceptual attack model where both users request for hardware A (with superior quality) but user-2 gets access to hardware B.

visibility about the hardware allocated to the program in the existing setup. This gives rise to following new challenges:

A. Hardware Suite

The cloud service provider may possess multiple hardware with varied degree of computing capability i.e., number of qubits and coupling map. For example, IBM Quantum has access to multiple quantum hardware like ibmq_london, ibmq_burlington, ibmq_essex, ibmq_santiago and ibmq_rochester (Fig. 1).

B. Multiple Choices for User-Specified Coupling Map

The scheduler at the cloud service provider end may have multiple hardware with identical coupling maps in the suite e.g., ibmq_london, ibmq_essex, ibmq_vigo etc. (Fig. 1). Structurally, one cannot differentiate them from each other.

C. Isomorphic Coupling Maps in Larger Hardware

The larger quantum hardware with more number of qubits possess multiple isomorphic coupling maps of many smaller hardware e.g., ibmq_rochester has many T-shaped coupling maps similar to ibmq_london as specified in one of the boxes in Fig. 1.

D. Quality and Cost Differences

Each of the hardware is distinct in terms of computation capability, quality of qubits, and cost. In general, the larger qubit hardware is more costly; the cost can depend on the qubit quality for identical hardware size.

E. Allocation/Scheduling Policy

The queue of quantum programs at the cloud end is typically long and maximizing the throughput is the key to reduce costs and help the scientific and industrial research communities in their explorations. The hardware scheduling policy typically employs a vendor-selected metric (throughput or first-in-first-out) to allocate the program to the hardware.

F. Validation of the Non-Repudiation of Results

The user cannot validate the origination of the result obtained from a quantum hardware. In other words, the user trusts the scheduler of the cloud provider to allocate the requested hardware to his workload.

Furthermore, quantum computers are being developed by multiple entities, some of which may be less trustworthy. In future, cloud-based quantum computing is expected to be offered by both trusted and less-trusted cloud vendors (that are located in less-trusted countries, for example). Performing reliable/trustworthy computing using these cloud-based quantum computers is an important step towards expanding their application space.

G. Proposed Attack Model

Consider the situation in Fig. 2 where User 1 (**U1**) and User 2 (**U2**) can access two quantum hardware **A** and **B** through the Cloud Service provider (**CS**) by paying certain service fee. Hardware A is relatively superior to B (in terms of error rates, for example), so both U1 and U2 would like to run their program on A. The motivation is to obtain high quality results as well as to reduce the cost if the problem is solved quickly. However, only one user can be assigned hardware A at a time i.e., U1 in this case. The scheduler in the CS has the option to make U2 wait or allocate it to B to maximize the throughput or increase user's cost (due to a malicious code segment or just to reduce the wait queue depth). As a result, U2 will suffer from poor quality/incorrect results due to inferior hardware and may also end up paying more. To address this problem, we propose a Quantum Physically Unclonable Function (QuPUF) program to be sent to the hardware to establish its identity first. The program will be allocated to the quantum computer (either the desired hardware or a different hardware) and the response will be sent back to the user. The user will match the response with the registered responses of the desired hardware. Since each quantum hardware has its own unique characteristics (e.g., single-/two-qubit gate error rates, decoherence and dephasing times), the responses of each hardware will be unique. Therefore, the user will be able to validate the identity of the hardware before sending his actual workload. The proposed QuPUF can be sent prior to the actual workload to establish trust (easy) or it can be embedded in

the user workload itself (complex, discussed in Section V). *To the best of our knowledge, this the first effort to establish the identity (trust) of a quantum computer*. Note that although we associated the proposed QuPUFs with an attack model, they can also be used as a standalone security and trust anchors.

H. Paper Contributions

We, (i) propose a new attack model and possible modes of attack; (ii) propose 2 flavors of QuPUFs to counter this attack model, (iii) study the stability of the QuPUF and propose to add/optimize the rotation angle of the QuPUF circuit to enhance the stability; (iv) introduce digitization of the response and optimized the bit-precision for improved inter- and intra-HD of the QuPUFs.

The paper is organized as follows. In Section II, we describe the background on quantum computing and PUF. The attack model is presented in Section III whereas the implementation details of various QuPUFs are provided in Section IV. The limitations of the proposed QuPUFs are discussed in Section V. Conclusion is drawn in Section VI.

II. Background

In this section, we discuss relevant background on quantum computing, PUF and quantum PUF.

A. Quantum Computing Concepts

1) Qubits: A qubit is the building block of a quantum computer. It stores data as quantum state. Unlike classical bits, qubit can be in a superposition state, i.e., a combination of 0 and 1 at the same time.

2) Quantum Gates: Quantum gates are the operations that modulate the state of qubits and thus, perform computations. Quantum gates can work on a single qubit (e.g., X (NOT) gate) or on multiple qubits (e.g., 2-qubit CNOT gates). Physically, they are realized using pulses (e.g., laser pulse in Ion Trap qubits, RF pulse in Superconducting qubit, etc.).

3) Errors in Noisy Quantum Computers: Present quantum computers suffer from various error modes such as, gate error, decoherence, readout error, single qubit error, two qubit error and crosstalk. Due to gate error the logical operation of a gate suffers certain probability of error. Qubits spontaneously interact with the environment and lose states which is known as decoherence. Due to imperfections in readout circuitry, qubits can suffer from bit-flips leading to readout errors. There are errors defined based on the type of gate. The errors caused by single qubit gates (like Hadamard gate for instance) is called single qubit error, and errors caused by two qubit gates (like CNOT gate) is called two qubit error. Finally, parallel gate operations on different qubits can affect each others' performance which is known as crosstalk. The rates of these errors vary among qubits and hardware which can be used as a signature to identify a particular hardware.

4) Various Factors Causing Errors: There could be many reasons behind errors including manufacturing imperfections, control error, thermal gradient, environmental interaction, poor microwave hygiene, etc [2]. Due to manufacturing imperfections, there can be defects/charge traps, and it leads to charge noise, which is a source of gate error. The control errors may stem from incorrectly calibrated gate pulses which may lead to under- or over-rotation of qubits, leakage to non-computation states, etc. For example, the quantum NOT (X) gate is realized by a 90° rotation around X-axis. A microwave pulse of certain amplitude, shape, and duration is applied to the qubit to drive this rotation. If the amplitude/shape/duration is incorrectly calibrated, the rotation will be less (under) or more (over) than the intended 90° leading to gate error. Qubits are ideally 2-level systems. the ground state (0) and 1^{st} excited state (1) make up the computational space. In Transmon qubits, there is a certain energy difference between 0 and 1 states usually denoted by a frequency f_{01}. A qubit is usually driven by a microwave pulse (gate pulse) of frequency f_{01} which will initiate transition between computational 0 and 1 states. However, in practical qubits like Transmon there are higher energy states like 2^{nd} excited state, 3^{rd} excited state, etc. beyond these two states. In case of Transmons, the energy difference between 1^{st} excited state and 2^{nd} excited state, f_{12}, is close to f_{01} (known as low anharmonicity). Due to this closeness of frequencies or low anharmonicity and imperfection in control signal, a qubit intended to be driven by f_{01} may jump out of 0 and 1 computational space and get excited to 2^{nd} excited state (known as leakage). Qubits are cooled down to cryogenic temperature and it is expected to have a homogenous temperature across the device. However, due to localized heating, there can be a thermal gradient. This thermal gradient is a source of decoherence [3]. Qubits are very susceptible to noisy environment like stray magnetic fields, heating, etc. For example, a qubit can absorb energy from environment and get excited to non-target state. Therefore, quantum computers are shielded and operated in very controlled environment. However, the solutions are not perfect yet and there are engineering challenges to prevent environmental effects completely. The microwave signal lines may suffer from photon number fluctuations [2] which causes stark shift. Due to stark shift, the operating frequency (f_{01}) of a qubit change. If the operating frequency of a qubit is different than the frequency it is driven, it gives rise to incorrect operation and gate error.

Errors (excluding crosstalk) are not dependent on the number of parallel operations, but rather on individual gate operations. Number of parallel operations give rise to crosstalk error. However, parallel single qubit gates do not incur significant crosstalk. Even this can be mitigated by serializing the gates e.g., by adding delays in the circuit like idle gates or barriers. Reliability is dependent on thermal variation rather than crosstalk. Due to this, the single qubit gate error changes over time, which gives rise to temporal variation, and contributes towards the intra-HD.

B. Physically Unclonable Function (PUF)

PUF [4] is a physical object which cannot be cloned. It acts a good security measure since the adversary cannot clone the characteristics of the PUF accurately. PUFs exploit the characteristics which are unique due to the variation in the manufacturing process. Some examples include SRAM

random initialization [5], thin-film resonators [6], dielectric properties of security coatings [7], delays in integrated circuits [8], etc. PUFs work on the principle of challenge and response. The user can provide a challenge to the PUF and obtain the corresponding response for authentication. The correctness of the response is validated by matching it from a database with registered challenges/response pairs (CRP) for the device. The PUFs can be categorized into based on the number of CRP, namely strong (exponential CRP e.g., arbiter PUF) and weak PUF (linear CRP e.g., SRAM PUF). Two important properties of PUF are, (i) inter-die Hamming Distance (HD) which measures the change in response between all pairs of identical chips for the same challenges.[1] The ideal value of inter-HD should be 50%; and, (ii) intra-die HD which measures the change in response with respect to time under temporal variation (due to noise, temperature and voltage fluctuations and aging). The ideal value of inter-HD should be 0%.

C. Quantum PUF

A common drawback of the classical PUFs is that the unique parameter created by the process variation is uncontrollable. As a result, even a slight change in the parameter cannot be reverted back to the original value. This can have an undesirable impact on the response for the challenges. Quantum PUFs have been proposed to address the above challenge by providing some controllable unique parameters. For instance, the concept of quantum confinement is employed to produce a unique signature by exploiting the fluctuations in quantum tunneling measurements inside resonant tunneling diodes (RTD) [9]. A quantum secure authentication (QSA) using illumination with a light pulse and checking the shape of the reflected light is also proposed [10]. However, both of these techniques heavily depend on quantum physics and are not applicable in the proposed application. A quantum challenge and quantum state readout of a classical optical PUF is proposed in [11]. Another work presents a quantum PUF which employs quantum properties of quantum device to establish a secure communication channel against quantum cryptographic attacks [12]. While some theoretical foundations are discussed, practical application, method, and circuits that relate to trustworthy computing in a cloud environment are completely unaccounted for.

D. Hardware Variability

In case of quantum computers, hardware variability manifests as variable "hardware errors", more specifically gate error rates, decoherence times (e.g., T1-relaxation), etc, across different quantum chips. In classical computing domain, two same chips may perform differently e.g., they may have different gate-delays due to manufacturing variations. This means hardware variability is manifested as variable gate-delays in classical chips. Likewise, variability between two quantum computing hardware is demonstrated/accounted as differences in gate error rates, decoherence (T1) times, etc. Therefore, similar to gate delays in classical chips - which can be used as a representation of variability and as a hardware signature – gate error-rates and decoherence times can be used as a representative of variability and hardware signatures in the quantum domain.

For example, the Transmon qubits used in the IBM machines is ultimately a solid-state device. "Trapped charge" in defects is a ubiquitous phenomenon in all solid-state devices alike. This trapped charge is a reason behind gate error in Transmon qubits [2]. Due to manufacturing variations, the number of defects will vary among qubits leading to variable gate errors which will be exploited to design the PUF.

III. Proposed Attack Model

In this section, we describe the attack model and various attack scenarios.

A. Basic Idea

In this attack model, the buggy or malicious or 3rd party controlled scheduler is the adversary which fails to allocate the quantum computer requested by the user's program but rather, (i) allocates a different quantum computer with identical coupling map, or (ii) maps the program to a smaller segment of a larger hardware along with other programs running in different segments in parallel. For example, the ibmq_rochester device in Fig. 1 (d) contains several T-shaped coupling maps (highlighted in red) where the programs meant for ibmq_london could be executed. This could be motivated by multiple types of scenarios described next. For simplicity, we call the quantum computer where the program was supposed to run as 'target' quantum computer and the one where the program actually runs as 'allocated' quantum computer.

B. Attack Scenarios

1) Scenario-1) Throughput Maximization: In this scenario, the scheduling routine attempts to make the best out of the hardware suite to maximize the throughput while adhering to the user-desired coupling map. This could either be intentional (a decision made by the cloud service provider) or due to a bug. If the user specified quantum computer is unavailable due to other prior tasks, the scheduler may divert this job to another quantum computer with the same or greater number of qubits. It is worth noting that if the number of qubits is same, then the computation will depend on the quality of the allocated hardware (which can be poor compared to the target hardware). However, if the number of qubits of the allocated quantum computer is higher than those of the target quantum computer, it is possible that another task is already being run on the allocated quantum computer. Such an arrangement, where two different tasks are being run simultaneously on the same quantum computer will give rise to crosstalk error due to inter-circuit interference [13], [14]. The quality of these subgraphs may be worse than the user specified hardware.

[1]In quantum context that we introduce next, these would be gate error, decoherence, readout error, and crosstalk.

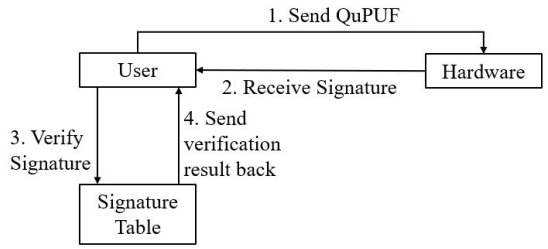


Fig. 3. The usage of QuPUF. Each step has been numbered.

2) Scenario-2) Low-Fidelity Allocation: In this scenario, the number of qubits of both the target and the allocated quantum computers are the same (possibly, the structure as well). However, the qubit quality defined by error rates like CNOT error rate and single-qubit U2 error rate and decoherence/dephasing is worse for the allocated quantum computer. Such an allocation can reduce the program fidelity. If the user is running hybrid algorithms, the poor fidelity outcome can increase the convergence time (i.e., number of iterations). By freeing up the queue, the adversary will also be able to improve throughput.

3) Scenario-3) Less-Trusted Quantum Computers: In future, less-trusted quantum computers could be available from 3rd parties that can allocate poor quality hardware and sabotage the output of the computing. Since the correct output of the optimization problem is not known, the user has to trust the sub-optimal result obtained from the quantum computer. In applications of national importance, this could have significant implication.

4) Scenario-4) Rogue Employee/Malicious Code in Scheduler: A rogue employee in trusted cloud vendor could try to sabotage the vendor's reputation by degrading the user compute fidelity just by tampering with the scheduling algorithm or rerouting the program to inferior hardware. Similar objectives can also be carried out if the scheduler is hacked by a malicious software. The rogue employee/malicious scheduler can also steal information by redirecting the programs to a 3rd party quantum hardware where they have full control.

C. Device Identification by QuPUF

Fig. 3 shows the steps involved in QuPUF based device authentication. It itvolves following phases:

1) Registration: First, a CRP database will be created similar to the conventional PUF (registration). For this step, the CRP of all qubits of each of the hardware will be collected and added to the CRP database. During validation the signature obtained from the hardware will be matched against the database for identification/validation.

2) Validation: The QuPUF, which is a quantum circuit, will be sent as a workload to the quantum hardware (step-1). Through this workload, the aim is to obtain the measurement results to act as the device signature. The expectation is that each hardware will produce a unique device signature depending on the internal characteristics such as, qubit quality through error rates, number of qubits, coupling map of the qubits, etc. Once the signatures have been obtained for each device (step-2), the user will query the CRP database (step-3) which will provide the hardwware corresponding to the signature (step-4). The user can then validate if the hardware is same as expected/requested.

IV. Quantum PUF and Experimental Results

In this section, we explain the proposed QuPUFs and present experimental results. We also compute the QuPUF quality and improve them using parametric rotation.

A. Generic Methodology and Experimental Setup

Each qubit of the quantum hardware is distinct in terms of 1-qubit and 2-qubit gate error, readout error, decoherence/dephasing and crosstalk error rates. The general strategy of QuPUF design is to convert these error rates into a qubit signature which in turn, will form the hardware signature. A very naive QuPUF could just initialize the qubits to ground state and perform a readout. It will convert the readout error into a signature. This paper only exploits the 1-qubit gate error, readout error and decoherence error to generate the signature although other means of designing the QuPUF are also possible. The proposed basic QuPUFs have single challenge and response (i.e., weak PUF) whereas resilient QuPUF employ rotation as a challenge. It is possible to expand the CRP by adding more challenges. ibmq_london, ibmq_burlington and ibmq_essex computers (Fig. 1 (a)) have been used for the basic QuPUFs. ibmq_london has been used for resilient Hadamard gate-based QuPUF and ibmq_vigo has been used for resilient decoherence-based QuPUF. For experiments with the QuPUFs, we have used real quantum hardware from IBM. For the basic QuPUFs, 75 experiments with 8192 shots per experiment were used, while for the resilient QuPUF, it was reduced to 20 experiments and 1024 shots per experiment due to long wait queue. The interval for measurements were also different for the basic and the resilient QuPUFs.

B. Basic QuPUFs

1) Hadamard Gate-Based QuPUF: This QuPUF exploits the biasing of the qubits towards 1 or 0 state to generate the response. The biasing could be a result of readout error (typically large) or the gate error (small for single qubit gates). Each qubit is initialized to zero state at the beginning. Next, the qubits are placed in a superposition state (using a Hadamard gate) followed by the measurement (Fig. 5(a)). Ideally, the qubits should produce equal probability of both 0 and 1 states. However, the probability is expected to be biased towards either zero or one, depending on the errors that will act as unique device signature.

2) Decoherence-Based QuPUF: This QuPUF exploits the differences in the decoherence times of the qubits to generate the response. The qubits are placed in an excited state and allowed to decohere for a fixed amount of time followed by the measurement operation. Some qubits decohere more and exhibit more 0 than 1 and vice versa is true for qubits that decohere less. The probability of 1 state acts as the unique response. We first initialize the qubits to ground state. Next,

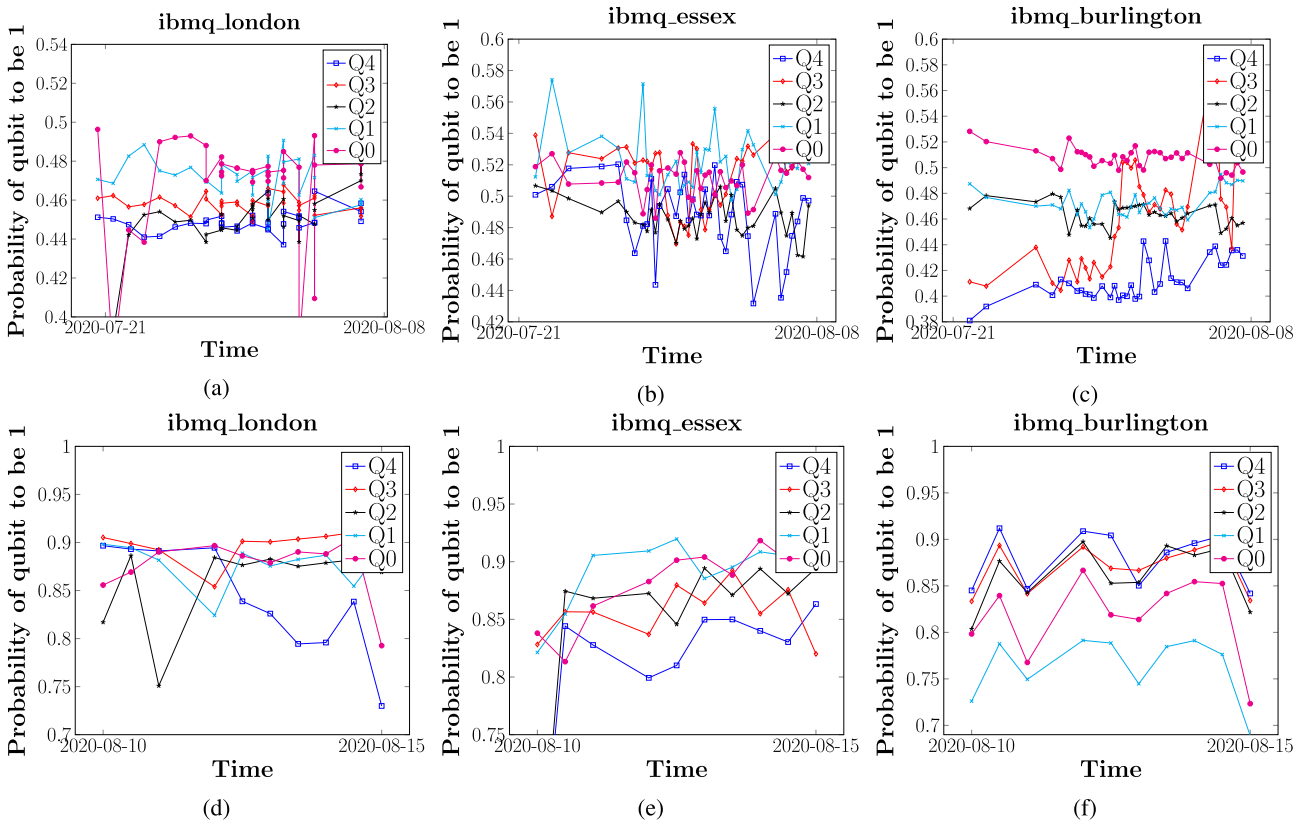


Fig. 4. Temporal variation in various quantum computers for the, (a)-(c) H-gate based; and (d)-(f) decoherence-based QuPUF.

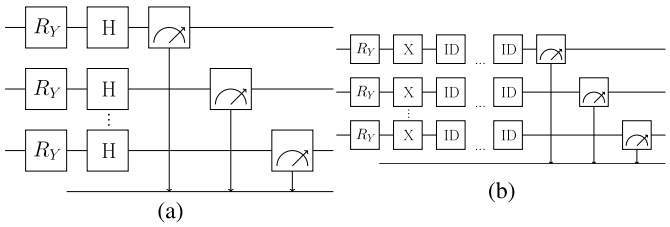


Fig. 5. Proposed QuPUFs: (a) Hadamard gate-based QuPUF; (b) decoherence-based QuPUF. The tunable rotation has been added for resilience.

we flip the state of the qubit using the X-Gate from 0 to 1 state. Finally, we allow it to decohere back to zero state by keeping the circuit idle i.e., by using idle gates followed by measurement (Fig. 5(b)).

3) Experimental Results: Fig 4 (a)-(c) shows the probability of 1 of the H-gate based QuPUF for the three hardware artifacts collected over few days. It can be observed that the probabilities of each qubit for all the quantum computers fluctuate significantly i.e., the device signature is sensitive to temporal variations. However, the signature also exhibits spatial variation which enables us to distinguish each qubit via the probability of 1. Fig. 4 (d)-(f) depicts the results of the three quantum computers for the decoherence-based QuPUF. Here, the spatial variation is more distinguished in terms of the mean separation, and the temporal variation is also relatively lower compared to the H-gate QuPUF. However, none of the qubits decohered to ground state implying that full decoherence did not occur due to less number of idle gates.

4) Interpretation of the Results: Fig 4 shows temporal variation of probability of 1 for various qubits in each of the three hardware for Hadamard gate-based and decoherence based PUFs. It clearly shows that absolute value of probability of 1 may not provide clear PUF signature. However, analytical techniques such as, mean and standard deviation of the PUF response can be used reliably. Fig 6 shows the boxplots for the resilient QuPUFs with varying rotation angle and varying number of idle gates. Here, the criteria for selection is to choose angle/ number of idle gates for every qubit which provide as much inter-qubit mean separation as possible, and also the least intra-qubit standard deviation.

For example, for Q0, we can select 4° and for Q1, we can select 5°. By selecting these values, we ensure that there is a mean separation (0.47 for Q0 and 0.43 for Q1) and the deviation is relatively less compared to other angles, implying that the variation of the selected angles is less.

C. Resilient QuPUFs

1) QuPUF With Tunable Rotation: The temporal variation of the quantum circuit is a function of the quantum state. It has been noted that adding parametric gate to the quantum circuit and tuning could optimize the resilience to dynamic variation [15]. Following similar line of thought, we added a tunable rotation gate (e.g., R_Y gate) to the H-gate and the X-gate in each qubit for resilience of the proposed QuPUFs to

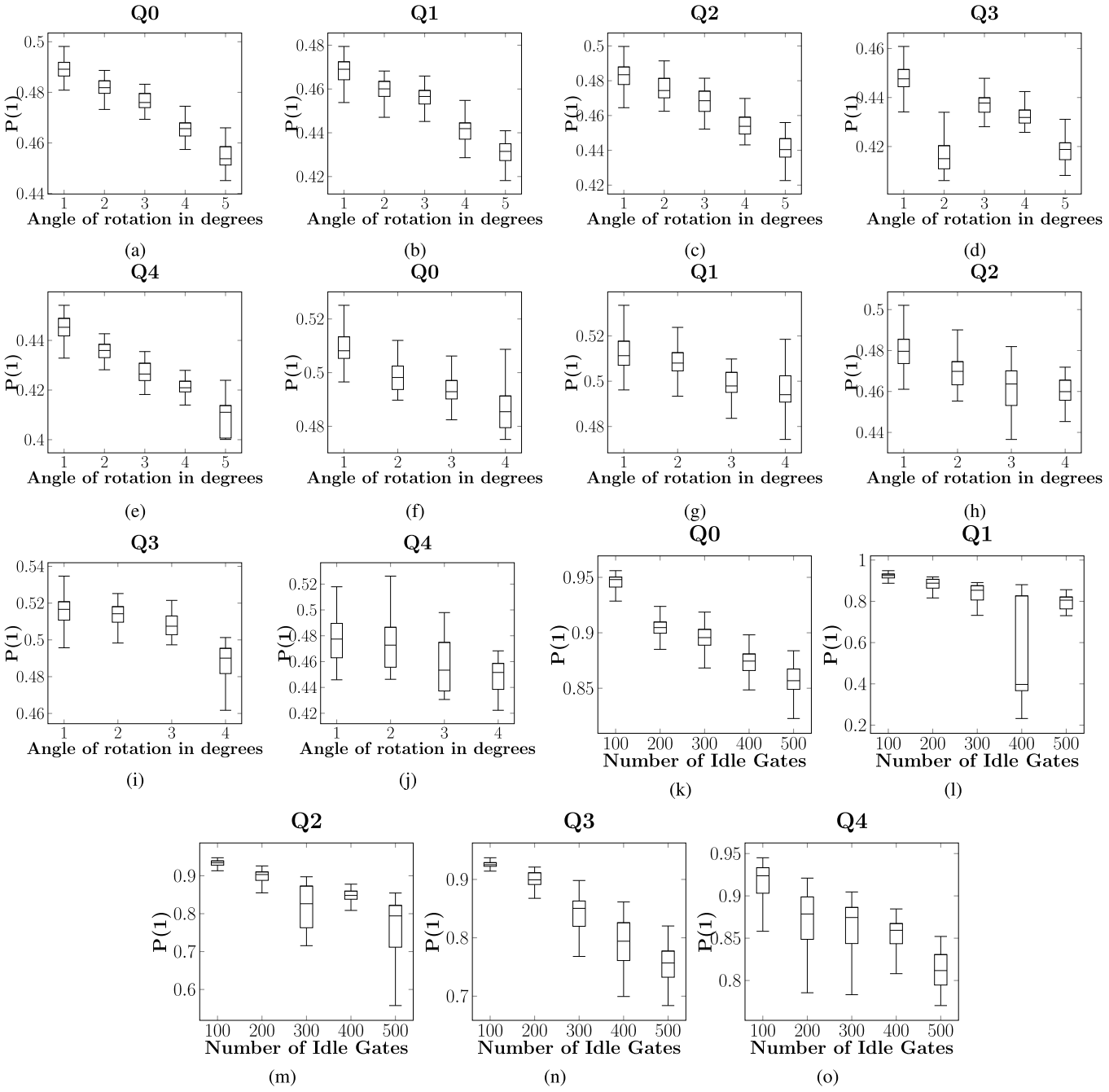


Fig. 6. Box plot of various qubits with different number of idle gates for **ibmq_london ((a)-(e)), ibmq_essex ((f)-(j)) and ibmq_vigo ((k)-(o))**. The sigma of the distribution minimizes with a specific number of idle gates.

temporal variation. The rotation angle could be varied slightly e.g., from 1° to 5° (Fig. 5). The tunable rotation can also act as a challenge to increase the challenge-response pair (CRP). A rotation towards 0 is expected to shift the probability of 1 towards the ground state. However, the inherent bias remains present for each quantum computer serving as a unique signature. For the decoherence based QuPUF, the number of idle gates has been varied to study the impact on stability for a fixed rotation angle. The rotation angle and the number of idle gates providing the optimal values of inter- and intra-HD are selected for the QuPUF (experimental HD results are described in Section IV-D).

2) Experimental Results: We sweep the rotation angle of H-gate based QuPUF and plot the mean and sigma of the probability of 1 for each qubit in Fig. 6. For the decoherence QuPUF, we sweep the number of idle gates from 100-500. Fig. 6 (k) - (o) shows the box-plots for each qubit with variable number of idle gates.

D. Quality Evaluation of the QuPUFs

1) HD Calculation: For calculating the intra- and inter-HD, we need to convert the analog value of hardware signature i.e., probability of '1' into a digital form. We split the range of probabilities in 32 steps for a 5-bit signature per qubit.

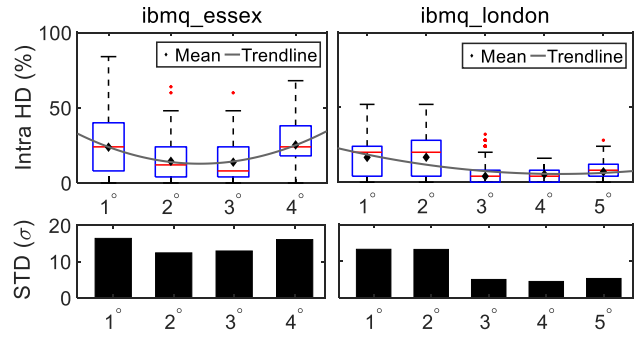


Fig. 7. Intra HD from two backends ibmq_essex and ibmq_london. X-axis shows the rotation angle θ in $R_Y(\theta)$ for H-gate based PUF. The trend line shows as we vary the angle, intra-HD also varies for an optimal angle it is lowest. For both ibmq_essex and ibmq_london, 3° shows the minimum intra-HD.

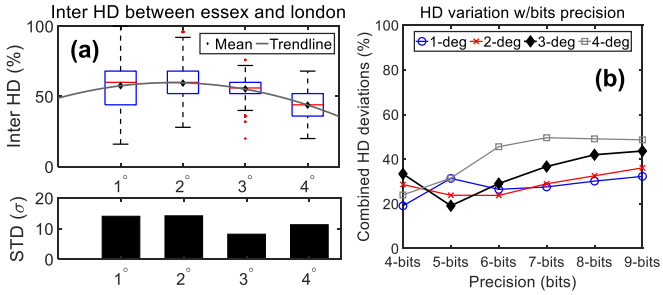


Fig. 8. (a) Inter-HD between ibmq_essex and ibmq_london; (b) combined inter- and intra-HD with varying bit precisions.

Thus, we get a 25 bit (5 qubit $\times$ 5 bits/qubit) signature for each data point for each of the 5 qubit hardware.

To compute intra-HD, we calculate the HD between all pairs of data points for the same quantum computer obtained over time and take the mean. To estimate the inter-HD, we compute the HD between all pairs of data-points for two different quantum computers and take the mean. The absolute value of the inter- and intra-HD will depend on the precision of the signature. We sweep the signature precision from 4-bit to 9-bit to choose the optimal value.

2) Experimental Results: Fig. 7 shows the intra-HD distributions from ibmq_essex and ibmq_london for 5-bits precision (H-gate based PUF with rotation). The top traces are the box-plots of distribution with explicitly plotted trend-lines of the means. The bottom traces show the standard deviations (σ). The plots depict that the intra-HDs vary with rotation angle ($RY(\theta)$) and exhibits an optima. For both ibmq_essex and ibmq_london, 3° is the optimal angle with lowest intra-HD (ibmq_essex 13.82% and ibmq_london 3.94%).

Fig. 8(a) shows the inter-HD between ibmq_essex and ibmq_london for 5 bits precision. This plot also shows inter-HD varies with rotation angle, and it is optimal for 3° (55.3%).

Finally, we plot both inter- and intra-HD with bit precisions and rotation angles (Fig. 8(b)). As we want inter-HD to be close to 50% and intra-HD close to 0%, a *combined HD deviation* metric is defined as $|inter\_HD-50|+|intra\_HD-0|$ which captures the deviations in both inter- and intra-HD. The lower combined value is desirable. We plot the results for inter-HD for ibmq_essex and ibmq_london and intra-HD from ibmq_essex. As we vary bit precision, the combined metric gives the optimal value for 3° rotation and 5−bits precision (optimal combined value 19.12%). Thus, setting angle and precision to 3° and 5-bits are best for QuPUF. Results with intra-HD from ibmq_london also shows similar behavior, and therefore, omitted for brevity.

3) Comparison Between QuPUFs: We also compute intra-HD for decoherence based PUF with data collected from ibmq_vigo with variable number of idle gates (100 – 400 idles). The intra-HD varies from 13% (100 idles) – 27% (400 idles) with a 5-bit precision. Therefore, a lower number of idle gate is better in decoherence-based PUFs in terms of intra-HD. Overall, H-gate based QuPUF performs better in terms of stability.

V. Discussions and Limitations

This section describes various aspects of the proposed QuPUF and potential limitations.

A. Ensuring Trust at Hardware Level

Validity and quality of cloud-based hardware providers is an important factor while establishing trust between the user and the hardware. However, it is also necessary to establish trust on hardware level itself. This is primarily because, (i) quantum hardware can provide high-quality signature like conventional CMOS PUFs and, (ii) it is ultimately the quality of the hardware that determines the accuracy of the result. An inferior hardware allocation can lead to higher costs and poor-quality solution which is undesirable. Hence, there is need of an assurance that the desired hardware is allocated to the user.

B. Bypassing QuPUF-Based Validation

It is possible that the QuPUF circuit is identified by the scheduler and routed to the correct hardware however, the actual user workload is rerouted to the incorrect hardware. This is possible if the vendor side scheduler is aware of the existence of QuPUF and employs a detection routine. This scenario can be addressed by embedding the QuPUF within the user workload. For example, user can validate the identity of few qubits while the other qubits are used for computation. This is possible for large workload running on large quantum hardware. For example, 3-4 qubits can be used to validate the identity of a 23-qubit hardware while the remaining qubits are used for computation. This approach will reduce the number of compute qubits. One can also use uncomputation to free up few qubits that have completed computation early and use them to run QuPUF circuit at no added overhead [16].

C. Other QuPUF Designs

Specific QuPUFs to exploit to readout error, 2-qubit gate errors and crosstalk can also be designed and evaluated for stability and uniqueness. It is also possible combine the responses of various QuPUFs to enhance the quality. For example, the response of H-gate and decoherence-based QuPUFs can be combined to identify the hardware more accurately than using them in isolation.

D. Challenge-Response Pairs (CRP)

The present implementation of the QuPUF is a weak PUF with one rotation gate and one rotation angle for each qubit. This can be expanded further by adding additional rotational gates to the already present R_Y gate, so that the challenge depends on more than one rotation angle. In such a scenario, the angle of rotation for each rotation gate can be considered a challenge. This approach will provide exponential CRP, with linear number of rotational gates (with rotations that provide stable HD).

E. Decohorence vs #Idle Gates

The intended effect of decoherence was not observed because the idle time produced by the idle gates was less than the coherence time (relaxation) of the hardware. The idle gates' purpose is to pass time in order to allow the qubits to relax to ground state. Since lesser number of idle gates were present in the circuit, the qubits did not get enough relaxation time, and as a result, did not decohere. This can be resolved by adding a greater number of idle gates. This requirement for higher number of idle gates was not supported until very recently by IBM systems.

F. Vulnerability to Temporal Variation

As seen from Fig. 4, the QuPUFs are sensitive to temporal variation. However, the intra-HD and inter-HD values obtained are satisfactory. This implies that even if one obtains dynamic hardware signatures, they can be identified since they are spaced out with respect to intra-HD and inter-HD. Also, the current trend shows that quantum industry is able to reduce the noise levels and increase the decoherence time aggressively. Therefore, the effectiveness of the proposed PUFs is expected to improve in future.

G. Unstable Decoherence Rates

The decoherence rates of modern NISQ computers are unstable, which poses a challenge for decoherence-based QuPUF. Varying decoherence rates will give varying amount of decoherence, and this will be reflected in the output. This might also cause increased readout error as measuring a qubit takes significantly longer than unitary operations on qubits, and during measurement, the qubits being measured may change their states due to decoherence [17]. Nevertheless, decoherence-based PUF is a potential direction to identify a quantum hardware once the variations are controlled at the hardware level.

H. Other Applications of the QuPUFs

The proposed QuPUFs can also be used to address other security challenges such as, Man-In-The-Middle (MITM) Attack. If the attacker tampers with the device signature it will be detected during the signature verification stage. The QuPUF signature can also be used for non-repudiation of data. For this application, the QuPUF signature will be appended with the results of a computation from a quantum computer to authenticate the computation outcome.

I. Comparison With Existing Remote Attestation Protocols

Note that the proposed QuPUF is a quantum-hardware security primitive that can be used as a building block of a security protocol (e.g., Intel SGX/TPM) to establish trust between - the user and the service provider - in a quantum-cloud computing environment. For instance, Intel SGX performs remote device attestation using Enhanced Privacy ID (EPID). EPID consists of the following four elements: member private key, group public key, message to be signed, and signature revocation proof list. In an SGX-like security platform for quantum-cloud, QuPUFs can be used to generate unique private keys for the quantum hardware (members). Although to the best of our knowledge, such security protocols are not in use today in the quantum-cloud computing platforms, we expect to see developments in this domain soon.

VI. Conclusion

We proposed two flavors of QuPUFs to establish trust in the public cloud-based quantum hardware. The proposed QuPUFs are thoroughly analyzed for uniqueness and stability on real quantum hardware. Our study indicated that minor tuning of parametric rotation of the QuPUF and choice of bit precision of the signature can optimize the response in presence of temporal and spatial variation in qubit quality. Experiments on real IBM quantum hardware show that the proposed QuPUF can achieve inter-die HD of 55% and intra-HD as low as 4%. The proposed QuPUFs can address wide range of security and trust issues associated with quantum computing.

As time progresses, the effectiveness of the proposed PUFs will improve due to sophisticated quantum control and temporal error mitigation efforts employed by quantum computing industry to improve the quality of the hardware.

Acknowledgment

The work is supported in parts by National Science Foundation (NSF) (CNS-1722557, CCF-1718474, OIA-2040667, DGE-1723687 and DGE-1821766) and seed grants from Penn State Institute for Computational and Data Sciences and Penn State Huck Institute of the Life Sciences. The authors thank the use of IBM Quantum Services for this work. The views expressed are those of the authors, and do not reflect the official policy or position of IBM or the IBM Quantum team. [18]

References

[1] P. Das, S. S. Tannu, P. J. Nair, and M. Qureshi, "A case for multi-programming quantum computers," in *Proc. 52nd Annu. IEEE/ACM Int. Symp. Microarchitecture*, Oct. 2019, pp. 291–303.

[2] P. Krantz, M. Kjaergaard, F. Yan, T. P. Orlando, S. Gustavsson, and W. D. Oliver, "A quantum engineer's guide to superconducting qubits," *Appl. Phys. Rev.*, vol. 6, no. 2, Jun. 2019, Art. no. 021318.

[3] S. Spilla, F. Hassler, and J. Splettstoesser, "Measurement and dephasing of a flux qubit due to heat currents," *New J. Phys.*, vol. 16, no. 4, Apr. 2014, Art. no. 045020.

[4] R. Pappu, B. Recht, J. Taylor, and N. Gershenfeld, "Physical one-way functions," *Science*, vol. 297, no. 5589, pp. 2026–2030, Sep. 2002.

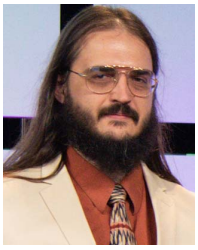
[5] J. Guajardo, S. S. Kumar, G.-J. Schrijen, and P. Tuyls, "FPGA intrinsic PUFs and their use for IP protection," in *Proc. Int. Workshop Cryptograph. Hardw. Embedded Syst.* Berlin, Germany: Springer, 2007, pp. 63–80.

[6] B. Skorić, T. Bel, A. Blom, B. de Jong, H. Kretschman, and A. Nellissen, "Randomized resonators as uniquely identifiable anti-counterfiting tags," in *Proc. Secure Compon. Syst. Identificat. Workshop*, Berlin, Germany, Mar. 2008.
[7] P. Tuyls, G.-J. Schrijen, B. Škorić, J. Van Geloven, N. Verhaegh, and R. Wolters, "Read-proof hardware from protective coatings," in *Proc. Int. Workshop Cryptograph. Hardw. Embedded Syst.* Berlin, Germany: Springer, 2006, pp. 369–383.
[8] B. Gassend, D. Clarke, M. van Dijk, and S. Devadas, "Silicon physical random functions," in *Proc. 9th ACM Conf. Comput. Commun. Secur. (CCS)*, 2002, pp. 148–160.
[9] J. Roberts *et al.*, "Using quantum confinement to uniquely identify devices," *Sci. Rep.*, vol. 5, no. 1, p. 16456, Dec. 2015.
[10] S. A. Goorden, M. Horstmann, A. P. Mosk, B. Škorić, and P. W. Pinkse, "Quantum-secure authentication of a physical unclonable key," *Optica*, vol. 1, no. 6, pp. 421–424, 2014.
[11] B. Škoric, "Quantum readout of physical unclonable functions," *Int. J. Quantum Inf.*, vol. 10, no. 1, Feb. 2012, Art. no. 1250001.
[12] M. Arapinis, M. Delavar, M. Doosti, and E. Kashefi, "Quantum physical unclonable functions: Possibilities and impossibilities," 2019, *arXiv:1910.02126*. [Online]. Available: http://arxiv.org/abs/1910.02126
[13] A. Ash-Saki, M. Alam, and S. Ghosh, "Analysis of crosstalk in NISQ devices and security implications in multi-programming regime," in *Proc. ACM/IEEE Int. Symp. Low Power Electron. Design*, Aug. 2020, pp. 25–30.
[14] P. Murali, D. C. Mckay, M. Martonosi, and A. Javadi-Abhari, "Software mitigation of crosstalk on noisy intermediate-scale quantum computers," in *Proc. 25th Int. Conf. Architectural Support Program. Lang. Operating Syst.*, Mar. 2020, pp. 1001–1016.
[15] M. Alam, A. Ash-Saki, and S. Ghosh, "Addressing temporal variations in qubit quality metrics for parameterized quantum circuits," in *Proc. IEEE/ACM Int. Symp. Low Power Electron. Design (ISLPED)*, Jul. 2019, pp. 1–6.
[16] Y. Ding *et al.*, "SQUARE: Strategic quantum ancilla reuse for modular quantum programs via cost-effective uncomputation," 2020, *arXiv:2004.08539*. [Online]. Available: http://arxiv.org/abs/2004.08539
[17] F. Leymann and J. Barzen, "The bitter truth about gate-based quantum algorithms in the NISQ era," *Quantum Sci. Technol.*, vol. 5, no. 4, 2020, Art. no. 044007.
[18] (2021). *IBM Quantum*. [Online]. Available: https://quantum-computing.ibm.com/

Koustubh Phalak received the bachelor's degree in electrical and electronics engineering from the Birla Institute of Technology and Science, Pilani, in 2020. He is currently pursuing the Ph.D. degree with the Department of Computer Science and Engineering, Pennsylvania State University. He works in the field of emerging technologies, especially quantum computing.

Abdullah Ash- Saki (Graduate Student Member, IEEE) received the bachelor's degree from the Bangladesh University of Engineering and Technology (BUET) in 2014. He is currently pursuing the Ph.D. degree with the Department of Electrical Engineering, Pennsylvania State University. He works on quantum computing. He was a recipient of Dr. Nirmal K. Bose Dissertation Excellence Award from the Department of Electrical Engineering, Penn State, the ECE Best Paper Award in 2020 ASEE Annual Conference, and Dr. Richard Newton Young Fellow Award in 55th Design Automation Conference (DAC).

Mahabubul Alam (Student Member, IEEE) received the B.Sc. degree in electrical and electronic engineering from the Bangladesh University of Engineering and Technology (BUET) in 2015. He is currently pursuing the Ph.D. degree in electrical engineering with Pennsylvania State University. Before joining the graduate school, he worked as an ASIC Physical Design Engineer with PrimeSilicon Technologies for over a year. He spent the summer of 2018 and 2020 as an Intern at Qualcomm and Intel, respectively. His current research interests include quantum computing, machine learning, design automation, and hardware security. He received the Best Paper Award (BPA) at ASEE (2020), BPA nomination at ISQED (2020), and secured second place at the Student Research Competition at ICCAD (SRC@ICCAD-2020).



Rasit Onur Topaloglu (Senior Member, IEEE) received the B.S. degree in electrical engineering from Bogazici University and the Ph.D. degree in computer science and engineering from the University of California at San Diego. He has worked for companies, such as Qualcomm, AMD, Globalfoundries. He is currently works as a Senior Hardware Developer with IBM, where he works on next-generation computer technology and design. He was partially involved with qubit characterization laboratory work at IBM Research. He has over 60 peer-reviewed publications and over 60 issued U.S. patents, more than a third of which are on quantum technologies. He has chaired the IEEE/ACM DAC Workshop on Design Automation for Quantum (DAQ). As of 2021, he is working on a *Quantum Computing* book. He serves on IEEE/ACM Design Automation Conference (DAC), IEEE/ACM International Conference on Computer-Aided Design (ICCAD), and IEEE International Symposium on Quality Electronic Design (ISQED) Technical Program Committees that cover quantum topics. He serves as the Chair for IEEE Mid-Hudson and the Secretary of ACM Poughkeepsie. He is an IEEE/ACM DAC Outstanding Innovator and an IBM Master Inventor.

Swaroop Ghosh (Senior Member, IEEE) received the B.E. degree (Hons.) from IIT, Roorkee, and the Ph.D. degree from Purdue University.

He is currently an Associate Professor with Pennsylvania State University. His research interests include quantum computing, emerging memory technologies, and hardware security.

Dr. Ghosh is a Senior Member of the National Academy of Inventors (NAI) and an Associate Member of Sigma Xi. He was a recipient of the Intel Technology and Manufacturing Group Excellence Award, the Intel Divisional Award, two Intel Departmental Awards, the USF Outstanding Research Achievement Award, the College of Engineering Outstanding Research Achievement Award, the DARPA Young Faculty Award (YFA), the ACM SIGDA Outstanding New Faculty Award, the YFA Director's Fellowship, the Monkowsky Career Development Award, the Lutron Spira Teaching Excellence Award, the Dean's Certificate of Excellence, and the Best Paper Award in American Society of Engineering Education (ASEE). He served as the General Chair, Conference Chair, and Program Chair of ISQED and DAC Ph.D. Forum and a Track (Co)-Chair of DAC, CICC, ISLPED, GLSVLSI, VLSID, and ISQED. He is a Distinguished Speaker of the Association for Computing Machinery (ACM). He served as an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS-I: REGULAR PAPERS and IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN. He served as a Senior Editorial Board Member of IEEE JOURNAL OF EMERGING TOPICS ON CIRCUITS AND SYSTEMS (JETCAS). He served as a Guest Editor of the IEEE JOURNAL OF EMERGING TOPICS ON CIRCUITS AND SYSTEMS (JETCAS) and IEEE TRANSACTIONS ON VLSI SYSTEMS. He has also served in the technical program committees of more than 25 ACM/IEEE conferences.