Fingerprinting Quantum Computer Equipment

Jalil Morris
Yale University
New Haven, CT, USA
jalil.morris@yale.edu

Anisul Abedin
Yale University
New Haven, CT, USA
anisul.abedin@yale.edu

Chuanqi Xu
Yale University
New Haven, CT, USA
chuanqi.xu@yale.edu

Jakub Szefer
Yale University
New Haven, CT, USA
jakub.szefer@yale.edu

ABSTRACT

With the increased real-world deployment of quantum computers, there is a security need to be able to fingerprint and track their equipment. This work proposes that cryogenic equipment used in superconducting qubit quantum computers could leverage inexpensive SRAM-based PUFs as fingerprints. This work is the first to perform a security evaluation of SRAM PUFs under cryogenic conditions using liquid nitrogen to rapidly freeze the memories to temperatures approaching −195℃ (−320°F or 77K). This work demonstrates that SRAM PUFs can become more stable under cryogenic conditions. As a result, a possible novel application of the SRAM PUFs is to identify and track quantum computer cryogenic hardware. Other means of fingerprinting quantum computer equipment are also possible, for example, based on the frequency of qubits. The ability to fingerprint quantum computers can be on one hand beneficial, to track the equipment, but on the other detrimental as attackers with access to the fingerprints could identify specific machines. Understanding the benefits and dangers of fingerprinting quantum computers, and securely deploying fingerprinting mechanisms is necessary to protect these emerging computing platforms.

CCS CONCEPTS

• **Security and privacy → Security in hardware**; • **Hardware → Quantum technologies**.

KEYWORDS

quantum computers, hardware security, fingerprinting

ACM Reference Format:
Jalil Morris, Anisul Abedin, Chuanqi Xu, and Jakub Szefer. 2023. Fingerprinting Quantum Computer Equipment. In *Proceedings of the Great Lakes Symposium on VLSI 2023 (GLSVLSI '23), June 5–7, 2023, Knoxville, TN, USA.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3583781.3590247

1 INTRODUCTION

A classic approach to device identification is to embed cryptographic keys in each device by burning them in at manufacturing time. However, this solution comes with potential pitfalls, such as increased production complexity as well as limited protection against key extraction attempts [1]. In order to address these issues, researchers have proposed a variety of Physical Unclonable Functions (PUFs).

Different types of PUFs exist, from intrinsic PUFs that leverage components already in the computer system, to extrinsic PUFs that require the addition of extra components where the PUF is located. Further, there are weak PUFs that have a limited number of challenge-response pairs (CRPs) and strong PUFs that aim to have the number of CRPs be exponential in the size of the PUF.

Among the different PUFs, the SRAM-based PUFs have been well studied and analyzed. They are weak PUFs, but have the benefit of being based on well-understood SRAM technology. There are also already commercialized and deployed in various devices, including in FPGAs for bitstream protection [18].

There is today, however, limited understanding of the behavior of the SRAM PUFs at extreme temperatures. In particular, most research on the reliability of SRAM PUFs focuses on elevated temperatures. Elevated temperatures can cause aging, and also many computer systems may want to run in elevated temperatures to save on energy, so it is important to know how SRAM PUFs behave at higher temperatures. On the other hand, there is limited data for SRAM PUFs at the other end of the spectrum: at extremely low temperatures. Extremely low temperatures can be induced as part of a cryogenic security attack [5] or naturally occur in novel but important computing settings such as cryogenic equipment used for quantum computers.

To help understand SRAM PUFs at extremely low temperatures, one contribution of this paper is to analyze SRAM PUFs at cryogenic temperatures using experiments with liquid nitrogen (LN_2). Especially, this paper evaluates the behavior of SRAM PUFs when liquid nitrogen is applied to rapidly freeze the memories to temperatures approaching −195℃ (−320°F or 77K). Understanding of the SRAM PUF operation at these temperatures enables the second contribution of this paper, which is the design of how cryogenic equipment used by quantum computers could be fingerprinted and tracked. Considering novel and important computing devices, such as superconducting qubit quantum computers, operate at extremely low temperatures, our results show that SRAM-based PUFs could be used to help aid to identify and fingerprint these computers or their components.

GLSVLSI '23, June 5–7, 2023, Knoxville, TN, USA
© 2023 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0125-2/23/06.
https://doi.org/10.1145/3583781.3590247

2 BACKGROUND AND RELATED WORK

This section provides a brief background on Physical Unclonable Functions (PUFs), in particular ones based on Static Random Access Memories (SRAM). This section also reviews work on cryogenic freezing of electronic components for analysis of their reliability and security. We are currently unaware of any work on fingerprinting and tracking cryogenic equipment, hence none of such work is listed in this section.

2.1 SRAM PUFs

A classic approach to computer device identification is to embed cryptographic keys in each device by burning them in at manufacturing time, e.g., using one-time fuses. However, this approach comes with potential pitfalls, such as increased production complexity as well as rather limited protection against key extraction attempts [1]. In order to address these issues, researchers have previously proposed Physical Unclonable Functions (PUFs). PUFs leverage the unique behavior of a device due to manufacturing variations as a hardware-based fingerprint. A PUF instance is assumed to be extremely difficult to replicate, even by the manufacturer. Since their initial proposal and development, PUFs have been proposed as cryptographic building blocks in security primitives and protocols for, among others, authentication and identification [15, 30, 32], hardware-software binding [9, 10, 16, 25, 26], remote attestation [17, 28], or secret key storage [33, 34].

Extrinsic types of PUFs in digital systems (such as arbiter PUFs [7, 30]) require the addition of dedicated circuits to the device and thus increase manufacturing costs and hardware complexity. Consequently, there is great interest in intrinsic PUFs [9], which are PUFs that are already inherent to a device. Intrinsic PUFs are considered an attractive low-cost security anchor, as they provide PUF instances within standard hardware that can be found in commercial off-the-shelf devices [23, 31], without requiring any hardware modifications. The most prominent example of an intrinsic PUF is a PUF based on Static Random-Access Memory (SRAM) [16, 20, 25, 27, 29], which draws its characteristics from the startup values of bi-stable SRAM memory cells. SRAM PUFs are known to have good PUF characteristics [14]. Intrinsic SRAM PUFs are mostly based on each SRAM cell's individual tendencies to either initialize to zero or one create a unique, hardware-based fingerprint for a given SRAM module. Recently, a new error-based SRAM PUF has been developed, which can be accessed at run-time [2].

This work focuses and extends research on the SRAM PUFs by analyzing how SRAMs behave under cryogenic conditions. We consider off-the-shelf SRAM modules and analyze the start-up values under cryogenic conditions, to the best of our knowledge, none of the prior works has explored this.

2.2 Cryogenic Security of SRAMs

This work focuses in part on understanding the behavior of SRAM and SRAM PUFs under freezing conditions. A number of works have recently been exploring cryogenic freezing as means of attacking electronic devices, in the so-called cryogenic security research area.

Prior security attacks based on rapidly cooling computer components are probably best exemplified by the Cold Boot attack [11], which focused on the internal capacitors found in the data cells of Dynamic Random Access Memory (DRAM) modules. The authors showed that by cooling DRAM chips, the decay rate of the capacitors used to store data bits in DRAMs is reduced. The cooling extends the time for which data persists in a powered-off chip, and allows malicious adversaries to transfer the cooled DRAM to a different computer to read the DRAM cells before the data is lost. The Cold Boot attacks focused on using up-side-down compressed air cans which leak compressed gas when used up-side-down and cool the DRAM. Authors in [11] also showed dipping DRAM chips in LN_2 to extend the time of the DRAM retention, but did not extensively evaluate DRAM chips under cryogenic conditions.

More recent freezing-based attack research demonstrated the Chill Out attack [13], which focused on the evaluation of the behavior of capacitors and DC/DC converters when exposed to cooling sprays. The authors showed that there is a decrease in capacitance when capacitors are cooled to about −55 °C, and that DC/DC converter behavior changes if the cooling is applied to the output electrolytic capacitors. The authors used off-the-shelf electronic cooling sprays and, similar to Cold Boot attack, up-side-down compressed air cans. Other recent work using off-the-shelf electronic cooling sprays [22] considered the security implications of freezing capacitors individually, and also freezing capacitors inside electronic filters and energy-storage capacitors in microcontrollers. Neither did the existing work evaluate using liquid nitrogen to bring these devices to extreme temperatures approaching −195 °C.

Most recently work [5] considered the security implications of ultra freezing of clock oscillator circuits. They demonstrated that MEMS-based clock oscillator circuits can have their output frequencies affected by the rapid freezing of the MEMS oscillators. The work demonstrated clock-glitching like behavior where the oscillator abruptly stops, then begins to operate again. Although extra care is needed to transport LN_2, the LN_2 attack cost can actually be lower than using cooling sprays or up-side-down compressed air cans. The authors discussed that cooling spray cans cost about $20 per 10 oz can, electronic duster compressed air cans cost about $5 per 3.5 oz can, while a liter of LN_2 needed for each attack costs about $0.10 when purchased in bulk.

Our work is motivated by the earlier freezing attacks, but we explore the effects of ultra freezing using LN_2 from a new perspective. We want to answer whether SRAM and SRAM PUFs are viable to operate at extremely low temperatures. We want to answer whether SRAM PUFs could be used to help aid to identify and fingerprint cryogenic quantum computer equipment, where devices are located at or below LN_2 temperatures.

3 FINGERPRINTING QUANTUM COMPUTERS AND THEIR CRYOGENIC EQUIPMENT

As one of our contributions, we propose a novel application of SRAM PUFs: as an additional chip used to help fingerprint and identify quantum computer cryogenic equipment (and any future controllers or classical computing equipment located inside the cryogenic chambers). As we outline below, there are rapid developments in quantum computers, yet there are today no mechanisms to fingerprint and identify quantum computer cryogenic equipment.

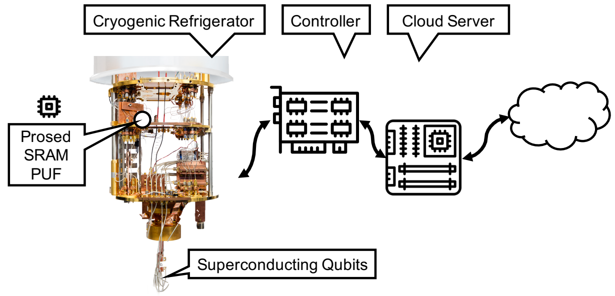

Figure 1: Simplified schematic of a modern quantum computer, including the cryogenic refrigerator for housing superconducting qubits, and external controllers and servers used to connect the quantum computer to the internet.

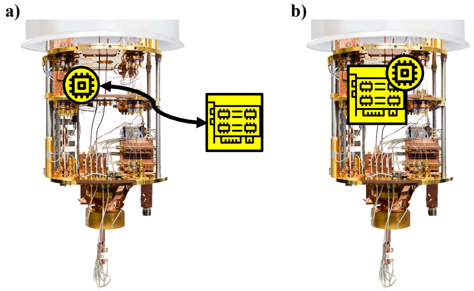

Figure 2: Two designs for SRAM PUF for use in cryogenic equipment: (a) inside the cryogenic refrigerator with an external controller and (b) inside the cryogenic refrigerator with an internal controller.

Adding an SRAM PUF chip to the equipment can enable tracking and identifying these expensive pieces of quantum computer hardware at a very low cost.

3.1 Brief Background on Quantum Computers

Today's quantum computers are commonly called Noisy Intermediate-Scale Quantum (NISQ) computers [24], as they are too small for quantum error correction (QEC) or even for large benchmarks, but they already show promising signs of useful applications in optimization, chemistry, and other important areas [12, 19, 21]. Further, quantum computing hardware keeps evolving at a fast pace, with 100-qubit quantum computers being now a reality and 1000-qubit quantum computers being projected to come online in the next few years [6]. Among different types of quantum computers, there are superconducting qubit quantum computers manufactured by IBM, Rigetti, and numerous other companies and startups. One key feature of these computers is the need for super-cold temperatures. A diagram of a typical superconducting qubit quantum computer is shown in Figure 1. For in-depth details about the design and operation of quantum computers, we refer the readers to existing textbooks, e.g., [4].

3.2 Securing Quantum Computers

The quantum computer is actually composed of many classical parts in addition to the actual chip where the qubits are realized, as shown in Figure 1. In particular, the qubits are housed inside the cryogenic fridge, but controlled by external, classical logic. In superconducting qubit quantum computers, the controller transmits and receives microwave pulses in order to operate and read out the qubits inside the fridge. The wires coming in and out of the cryogenic fridge are one potential bottleneck in the design and future scaling of quantum computers. As a result, preliminary research has begun to explore placing some (or even all) of the control logic inside the cryogenic fridge [3].

The cryogenic fridge, however, forms a natural boundary against physical analysis and inspection. It is difficult to identify the hardware inside the fridge, unless it is warmed up and disassembled. Unfortunately, warming up the fridge not only obviously raises the temperature of the components, but breaks vacuum seals inside, and in some cases may even physically damage the superconducting chips. As such we believe there is a need for a physical component, or a hardware anchor, that can be used for identification and fingerprinting that is inside the fridge and can be used at run-time without the need to warm up and open up the fridge.

The research presented in this paper for the first time analyzed the SRAM PUFs at cryogenic temperatures, and given the favorable results, we believe then that a new logical application of SRAM PUFs is to identify and fingerprint quantum computers.

3.3 A Design for SRAM-based Fingerprinting for Use in Cryogenic Equipment

We propose two designs for SRAM PUF for use in cryogenic refrigerators: stand-alone and with a controller are shown in Figure 2. Design in part a) of the figure corresponds to the deployments today, while part b) anticipates future deployments where the controller is inside the fridge.

3.4 Threat Model

We assume that a dedicated SRAM chip is added inside the cryogenic equipment. We assume the internal controller driving the power and control signals to trigger the SRAM and read out the start-up values of the SRAM which form the PUF. Existing work already explores how classical digital devices, such as controllers, could operate at cryogenic temperatures [3]. We assume the cryogenic equipment forms a natural boundary for probing and analyzing the internal components. As result, we assume that the SRAM chip cannot be manipulated or removed by an adversary.

3.5 Applications of SRAM PUFs to Cryogenic Equipment and Quantum Computers

We propose three use cases for SRAM PUFs. First, identification and fingerprinting of quantum computer hardware. Second, protection of control algorithms (when combined with a controller that is also inside the cryogenic refrigerator). Third, aging-based tracking of the use of quantum computers.

First, in the case of SRAM PUF inside the cryogenic refrigerator, the PUF can be used to identify the quantum computer hardware. It could be used to identify the particular quantum computer by having the controller read out the PUF startup state. If the owner of the quantum computer is not trusted, then this would make it

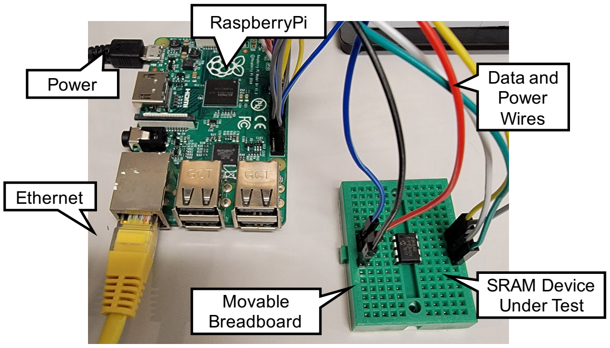

Figure 3: Experimental setup.

possible for the owner to clone the PUF very easily. A controlled PUF interface [8] and associated logic may have to be included inside the cryogenic fridge to prevent cloning of the PUF.

Second, for the case of SRAM PUF inside the cryogenic refrigerator as part of a future design that has other quantum computer control logic inside the cryogenic refrigerator as well, the SRAM PUF could also be used for encryption and decryption of data. The SRAM PUF could in such scenarios also be used for bitstream encryption to protect the proprietary control algorithms being loaded onto the FPGA. Since the controller is already inside the fridge in this scenario, it could be used to provide the controlled PUF interface so that raw PUF values are never sent outside of the fridge as well.

Third, we also envision aging-based tracking of quantum computers. With additional logic inside the fridge (either added hardware component in the first use case or using the controller in the second use case) the SRAM cloud be activated as the quantum computer runs. SRAM activation could be used to induce aging in the SRAMs, which could be measured by analyzing the changes in the startup values of the SRAM PUF. In addition, if the FPGA controller is in the cryogenic fridge, the aging analysis could be performed on the SRAMs that are part of the FPGA, thus not requiring additional SRAM components to be installed.

4 EVALUATION

To understand if the SRAMs and SRAM PUFs are viable under cryogenic conditions such as inside cryogenic refrigerators, this work evaluates the behavior of SRAM PUFs under freezing using liquid nitrogen. In particular, the evaluation approach leverages liquid nitrogen, LN_2, to rapidly freeze the SRAM memories to temperatures approaching −195 °C (−320 °F or 77 K). The freezing is achieved by pouring LN_2 from a dewar onto the target device for varying amounts of time to explore how the freezing affects the SRAM startup values.

The SRAMs are controlled by a RaspberryPi board, as shown in Figure 3. The SRAM modules, listed in Table 1 are connected to the RaspberryPi via GPIO pins available on the RaspberryPi board. The SRAMs are located on a small breadboard and connected via flexible jumper wires as shown in Figure 3. The breadboard can be placed in an insulated container so that the LN_2 can be poured on the SRAM memory, while the RaspberryPi remains outside and is not impacted by the freezing.

Table 1: SRAM memories used in the evaluation.

Manufacturer	Model Number	Size (KB)
Microchip	23K640-E/P	64 µF
Microchip	23K256-E/P	256 µF
Microchip	23LCV512-I/P	512 µF

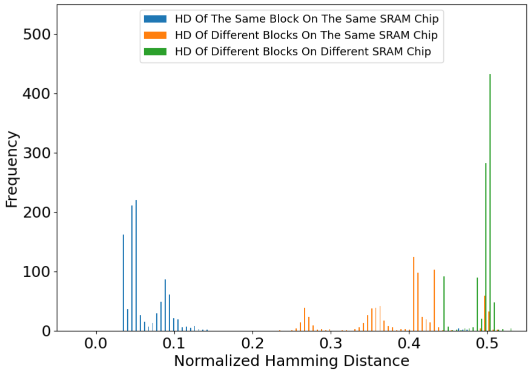

Figure 4: Comparison of inter- and intra-Hamming distances of SRAM PUFs startup values without freezing.

Each SRAM was divided into 64 KB regions, each forming a logical PUF. Note that the smallest SRAM modules used could only fit one SRAM logical PUF. The RaspberryPi was used to turn the memories off and on, and then read out the startup value of the SRAM logical PUF regions. The RaspberryPi was controller over the Ethernet and was running code that turned the SRAMs off and on at fixed intervals so that the startup values, or the PUF readouts, could be made at fixed intervals while LN_2 was applied. The SRAM startup values were saved in RaspberryPi's memory that were not subject to freezing.

4.1 Baseline Under Normal Conditions

We first evaluated the SRAM memories without freezing to establish whether the inter- and intra-Hamming distance of these particular memories made them suitable for PUFs. As shown in Figure 4, there is indeed a clear separation between the inter- and intra-Hamming distance of these particular memories. The blue bars show the intra-distance, with about 10% or less variation in the startup values. The orange and green bars show the inter-distance. Interestingly the distance between logical PUFs on the same physical SRAM (orange bars) is more than among logical PUFs on different SRAMs (green bars). Based on our later analysis, we believe this is due to possible temperature variation when data used to generate this graph was collected. Nevertheless, this clear separation between inter- and intra-Hamming distances even with larger than expected variations in the inter-Hamming distances.

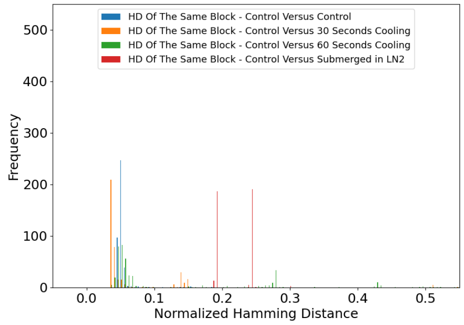

Figure 5: Comparison of Hamming distances of SRAM PUFs startup values without freezing compared to startup values with freezing.

4.2 Freezing vs. No Freezing

We next evaluated the SRAM startup value-based PUFs with 30 seconds and 60 seconds of LN_2 pouring, as well as dipping the SRAM modules on LN_2. We analyzed the Hamming distance of the startup values of each logical SRAM PUF block to itself under different conditions. The results are shown in Figure 5. The figure reveals that as Hamming distance between a block and itself without freezing is as expected very close. Interestingly, when the SRAM is frozen, the distance between the block (unfrozen) and itself (frozen) increases, meaning the startup values of SRAM PUF enrolled at room temperature may give much different readout values when queried at LN_2 temperatures. Considering the intra-Hamming distance (blue bars) we believe that non-uniform freezing when LN_2 is being poured contributes to the large distribution of the values.

4.3 Analysis Under Cryogenic Conditions

Figure 6 shows the evaluation of inter- and intra-Hamming distances when the logical SRAM PUF blocks were frozen. The intra-Hamming distance (shown in blue bars) has a clear separation from the inter-Hamming distance (shown in orange and green bars).

We believe that the results of these and the freezing vs. no freezing experiments indicate two characteristics of SRAM PUFs under cryogenic conditions. First, cryogenic freezing stabilizes the PUFs and limits thermal and other random noises. Second, the startup values for the frozen SRAM PUFs are slightly different from the startup values for the same SRAM PUFs at room temperature.

4.4 Analysis of Freezing Timing Impact

Next we evaluated how the freezing time affects the SRAM PUF value. We computed Hamming distances between the startup values of each of the different logical SRAM PUFs to itself, but when the freezing timing was different. Figure 7 shows the results. Blue bars are the comparison of logical SRAM PUF blocks to themselves. The orange bars compare the blocks to themselves when exposed to 30s of LN_2 pouring. The green bars compare the blocks to themselves when exposed to 60s of LN_2 pouring. The red bars compare the blocks to themselves when submerged in LN_2.

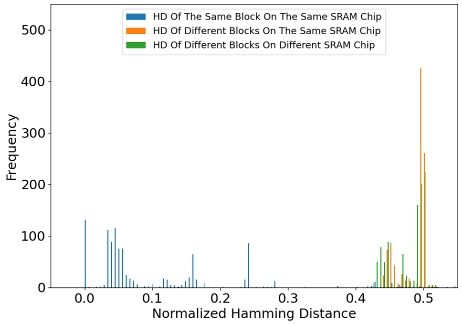

Figure 6: Comparison of inter- and intra-Hamming distances of SRAM PUFs under freezing conditions.

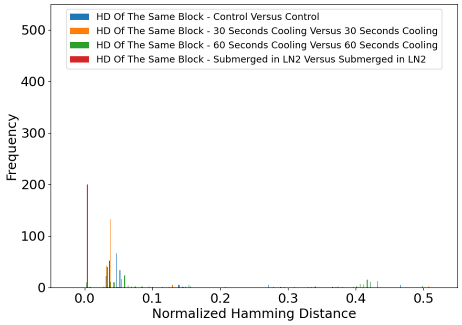

Figure 7: Comparison of inter-Hamming distances of SRAM PUFs under freezing conditions.

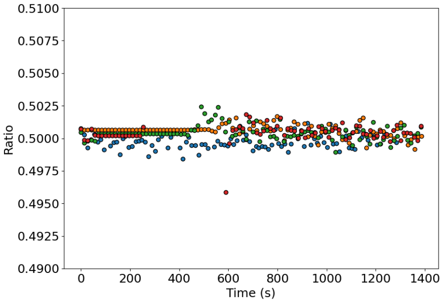

Figure 8: Ratio of startup bits starting in 1 state for different freezing and thawing times. Blue dots show no freezing, red dots show 30s freezing, green dots show 60s freezing, and orange show dipping in LN_2.

From the results, we can see that submerged SRAM PUFs give the most stable and consistent readouts (smallest Hamming distances). Meanwhile, the other measurements have some deviations, with some green bars having quite large Hamming distances. We believe this matches with our experimental observation that pouring of LN_2 is not precise as the liquid splashes around the container and may or may not equally cool different parts of the SRAM chips. As expected, submerging SRAMs in LN_2 gives the most uniform and stable freezing conditions leading to very nice and very low Hamming distances.

Lastly, we measured the ratio of startup bits starting in state 1 for different freezing and thawing times, as shown in Figure 8. Without freezing (blue dots) there is a rather constant distribution of ratios, with a startup value of 1 being very slightly less than

50%. With different amounts of freezing, the ratio is very stable, but slightly above 50%. Note that the freezing effects last longer than the application of LN_2, i.e. even if the pouring of LN_2 is stopped, the SRAM remains frozen for some time. After the freezing is stopped, eventually the ratio again returns to a distribution of values.

4.5 Summary of Results

Our experiments have shown that SRAM PUFs can be operated under cryogenic conditions. We have shown that SRAM PUFs are stable under LN_2 freezing, and thus could be considered to be placed inside the cryogenic equipment of quantum computers. Especially interestingly, the most stable behavior is when SRAM chips are fully submerged in LN_2, which best models the conditions at the top parts of the cryogenic refrigerator used by superconducting qubit quantum computers.

5 CONCLUSION AND FUTURE WORK

This work proposed a novel application of SRAM PUFs, which is to identify and track quantum computer hardware operating in cryogenic fridges. To understand the viability of SRAMs and SRAM PUFs under extremely cold conditions, this work leveraged liquid nitrogen to rapidly freeze the SRAM memories to temperatures approaching −195 °C (−320 °F or 77 K). Following the evaluation, this work demonstrated that SRAM PUFs can actually be more stable under cryogenic conditions. While the pre- and post-freezing readout of the SRAM PUFs changes, the post-freezing digital fingerprints are stable and exhibit good inter- and intra-distances indicating that they can be used for digital fingerprints under cryogenic conditions.

As future work, further evaluation of SRAM PUFs can be done for longer periods of time, or at even lower temperatures. Given the constant operation of quantum computers are these very low temperatures, SRAM PUFs could be analyzed after weeks or months of freezing. Further, modern dilution refrigerators used in quantum computers have parts that reach temperatures as low as 2 mK (this is where the qubits are located). While the upper stages are at about 77 K which is the liquid nitrogen temperature, the SRAM PUFs could be evaluated at these even more extreme conditions.

ACKNOWLEDGMENT

The authors would like to thank Kelly Woods and the Yale University cleanroom for assistance in obtaining liquid nitrogen used in the experiments. This work was supported in part by NSF grant 1901901.

REFERENCES

[1] Frederik Armknecht, Roel Maes, Ahmad-Reza Sadeghi, Berk Sunar, and Pim Tuyls. 2010. Memory leakage-resilient encryption based on physically unclonable functions. In *Towards Hardware-Intrinsic Security*. Springer, 135–164.

[2] Anys Bacha and Radu Teodorescu. 2015. Authenticache: harnessing cache ECC for system authentication. In *Proceedings of International Symposium on Microarchitecture*. ACM, 128–140.

[3] ID Conway Lamb, JI Colless, JM Hornibrook, SJ Pauka, SJ Waddy, MK Frechtling, and DJ Reilly. 2016. An FPGA-based instrumentation platform for use at deep cryogenic temperatures. *Review of Scientific Instruments* 87, 1 (2016), 014701.

[4] Yongshan Ding and Frederic T Chong. 2020. Quantum computer systems: Research for noisy intermediate-scale quantum computers. *Synthesis Lectures on Computer Architecture* 15, 2 (2020), 1–227.

[5] Jonathon Durand, Anisul Abedin, and Jakub Szefer. 2021. Ultra Freezing Attacks and Clock Glitching of Clock Oscillator Circuits. In *Proceedings of the Asian Hardware Oriented Security and Trust Symposium (AsianHOST)*.

[6] Jay Gambetta. 2020. IBM's Roadmap For Scaling Quantum Technology.

[7] Blaise Gassend, Dwaine Clarke, Marten Van Dijk, and Srinivas Devadas. 2003. Delay-based circuit authentication and applications. In *Proceedings of the ACM Symposium on Applied Computing*. ACM, 294–301.

[8] Blaise Gassend, Marten Van Dijk, Dwaine Clarke, Emina Torlak, Srinivas Devadas, and Pim Tuyls. 2008. Controlled physical random functions and applications. *ACM Transactions on Information and System Security (TISSEC)* 10, 4 (2008), 1–22.

[9] Jorge Guajardo, Sandeep S Kumar, Geert-Jan Schrijen, and Pim Tuyls. 2007. *FPGA intrinsic PUFs and their use for IP protection*. Springer. 63–80 pages.

[10] Jorge Guajardo, Sandeep S Kumar, Geert Jan Schrijen, and Pim Tuyls. 2008. Brand and IP protection with physical unclonable functions. In *IEEE International Symposium on Circuits and Systems*. 3186–3189.

[11] J Alex Halderman, Seth D Schoen, Nadia Heninger, William Clarkson, William Paul, Joseph A Calandrino, Ariel J Feldman, Jacob Appelbaum, and Edward W Felten. 2009. Lest we remember: cold-boot attacks on encryption keys. *Commun. ACM* 52, 5 (2009), 91–98.

[12] Jonathan A Jones, Michele Mosca, and Rasmus H Hansen. 1998. Implementation of a quantum search algorithm on a quantum computer. *Nature* 393, 6683 (1998), 344–346.

[13] Obi Nnorom Jr., Jalil Morris, Ilias Giechaskiel, and Jakub Szefer. 2021. Chill Out: Freezing Attacks on Capacitors and DC/DC Converters. In *European Test Symposium (ETS)*.

[14] Stefan Katzenbeisser, Ünal Kocabaş, Vladimir Rožić, Ahmad-Reza Sadeghi, Ingrid Verbauwhede, and Christian Wachsmann. 2012. PUFs: Myth, fact or busted? A security evaluation of physically unclonable functions (PUFs) cast in silicon. In *Cryptographic Hardware and Embedded Systems*. Springer, 283–301.

[15] Ünal Kocabaş, Andreas Peter, Stefan Katzenbeisser, and Ahmad-Reza Sadeghi. 2012. *Converse PUF-based authentication*. Springer.

[16] Florian Kohnhäuser, André Schaller, and Stefan Katzenbeisser. 2015. PUF-Based Software Protection for Low-End Embedded Devices. In *Trust and Trustworthy Computing*. Springer, 3–21.

[17] Joonho Kong, Farinaz Koushanfar, Praveen K Pendyala, Ahmad-Reza Sadeghi, and Christian Wachsmann. 2014. PUFatt: Embedded platform attestation based on novel processor-based PUFs. In *ACM/EDAC/IEEE Design Automation Conference*. 1–6.

[18] Sandeep S Kumar, Jorge Guajardo, Roel Maes, Geert-Jan Schrijen, and Pim Tuyls. 2008. The butterfly PUF protecting IP on every FPGA. In *2008 IEEE International Workshop on Hardware-Oriented Security and Trust*. IEEE, 67–70.

[19] Benjamin P Lanyon, James D Whitfield, Geoff G Gillett, Michael E Goggin, Marcelo P Almeida, Ivan Kassal, Jacob D Biamonte, Masoud Mohseni, Ben J Powell, and Marco Barbieri. 2010. Towards quantum chemistry on a quantum computer. *Nature chemistry* 2, 2 (2010), 106–111.

[20] Roel Maes, Vladimir Rožić, Ingrid Verbauwhede, Patrick Koeberl, Erik Van der Sluis, and Vincent Van der Leest. 2012. Experimental evaluation of Physically Unclonable Functions in 65 nm CMOS. In *Proceedings of the ESSCIRC*. 486–489.

[21] N David Mermin. 2007. *Quantum computer science: an introduction*. Cambridge University Press.

[22] Jalil Morris, Obi Nnorom Jr., Anisul Abedin, Ferhat Erata, and Jakub Szefer. 2021. Deep Freezing Attacks on Capacitors and Electronic Circuits. In *Proceedings of the International Conference on Security, Privacy and Applied Cryptographic Engineering (SPACE)*.

[23] phoneAsToken [n.d.]. SBIR project: Bring Your Own Security, (Online). https://www.dcypher.nl/files/Intrinsic-ID.pdf last accessed, 08.07.2016.

[24] John Preskill. 2018. Quantum computing in the NISQ era and beyond. *Quantum* 2 (2018), 79.

[25] André Schaller, Tolga Arul, Vincent van der Leest, and Stefan Katzenbeisser. 2014. Lightweight Anti-counterfeiting Solution for Low-End Commodity Hardware Using Inherent PUFs. In *Trust and Trustworthy Computing*. Springer, 83–100.

[26] Ryan A Scheel and Akhilesh Tyagi. 2015. Characterizing Composite User-Device Touchscreen Physical Unclonable Functions (PUFs) for Mobile Device Authentication. In *Proceedings of the International Workshop on Trustworthy Embedded Devices*. ACM, 3–13.

[27] Geert-Jan Schrijen and Vincent van der Leest. 2012. Comparative analysis of SRAM memories used as PUF primitives. In *Proceedings of the Conference on Design, Automation and Test in Europe*. EDA Consortium, 1319–1324.

[28] Steffen Schulz, Ahmad-Reza Sadeghi, and Christian Wachsmann. 2011. Short paper: lightweight remote attestation using physical functions. In *Proceedings of the ACM Conference on Wireless Network Security*. 109–114.

[29] Georgios Selimis, Mario Konijnenburg, Maryam Ashouei, Jos Huisken, Harmke De Groot, Vincent Van der Leest, Geert-Jan Schrijen, Marten Van Hulst, and Pim Tuyls. 2011. Evaluation of 90nm 6T-SRAM as Physical Unclonable Function for secure key generation in wireless sensor nodes. In *IEEE International Symposium on Circuits and Systems*. 567–570.

[30] G Edward Suh and Srinivas Devadas. 2007. Physical unclonable functions for device authentication and secret key generation. In *Proceedings of the Design Automation Conference*. 9–14.

[31] trustedSensor [n.d.]. Intrinsic-ID to Showcase TrustedSensor IoT Security Solution at InvenSense Developers Conference. https://www.intrinsic-id.com/intrinsic-id-to-showcase-trustedsensor-iot-security-solution-at-invensense-developers-conference/, accessed Feb. 2016.

[32] Pim Tuyls and Lejla Batina. 2006. RFID-tags for Anti-Counterfeiting. In *Topics in Cryptology*. Springer, 115–131.

[33] Pim Tuyls, Geert-Jan Schrijen, Frans Willems, Tanya Ignatenko, and B Skoric. 2007. Secure key storage with PUFs. *Security with Noisy Data–On Private Biometrics, Secure Key Storage and Anti-Counterfeiting* (2007), 269–292.

[34] Pim Tuyls and Boris Škorić. 2006. Secret key generation from classical physics: Physical uncloneable functions. In *AmIware Hardware Technology Drivers of Ambient Intelligence*. Springer, 421–447.