LEIDEN UNIVERSITY

Risk management and the quantum threat

Understanding the requirements to run Shor's algorithm to break RSA with a 2048 bit key and how to use this information to protect against the quantum threat

by

ir. C. Rugers

A thesis submitted in partial fulfillment for the
degree of Master of Science

at the
Cyber Security Academy

January 2018

Examination Committee:
prof. dr. ir. J. van den Berg,
dr. C. Schaffner and
ir. R. Versluis

"If you think you understand quantum mechanics, you don't!"

attributed to R.P. Feynman.

Abstract

Quantum computers are supposed to be of great value for research in the field of medicine, material science, energy, transport and logistics. However several standardization institutes like NIST report that the quantum computer poses a threat to cyber security. The impact of quantum computing on broadly deployed crypto-systems is clear. Shor's algorithm will break public-key-cryptography systems and Grover's algorithm will weaken symmetric-key-cryptography systems. However, the time of arrival of a quantum computer that can run these algorithms is not as clear as the impact. This thesis investigated the requirements for running Shor's algorithm to break RSA-2048 and uses the obtained information to propose tools that support risk-management regarding the quantum threat.

We conclude that the gap between the requirements to run Shor's algorithm to break RSA-2048 and the available physical resources is significant. This gap can be closed by progress on the supply side and by progress on the demand side. On the demand side, progress in fault-tolerant architectures and/or reducing the required number of T-gates will significantly reduce the gap. On the supply side, progress in the number of physical controllable qubits and an increase in the fidelity of quantum gates will significantly reduce the gap. Also should be noted that breaking other algorithms requires a different set of physical resources, which results in other gaps.

This information is used to define the topics to monitor the quantum threat regarding its impact on cyber security. Monitoring is necessary because our society depends heavily on ICT. The vulnerable crypto-systems are used in various types of security protocols, which facilitate services like online shopping and banking, online access to your health-care test results, online registration for social funds for citizens, providing citizens online trusted information about calamities. Not being able to trust these technologies supporting these and other online services cripples society.

The government has its responsibilities regarding cyber security as well as businesses and organizations as described in the second national cyber security strategy. The national cyber security center can use its role as expert authority on cyber security to provide public and private parties with information derived from monitoring the quantum threat. For this purpose, the proposed monitoring framework can be used. Businesses and organizations can use the proposed translation method to determine how the quantum threat impacts their strategic risks.

Additionally a pragmatic approach is proposed. It uses the results of the proposed translation method to select one of three scenarios. Each scenario has guidelines for actions depending on the significance of the impact on strategic risks. Using the pragmatic approach creates a balance between the uncertainty about when the risk materializes and the investments required to investigate and act upon the risk. This is different compared to other approaches, which use an asset-based approach and start prioritizing after an inventory of vulnerable IT assets.

The monitoring framework, the translation method and the pragmatic approach give the public and private sector tools to act on the quantum threat. This is a first step in enabling society to reduce the negative consequences of quantum computing on society and to fully benefit from positive consequences.

Acknowledgements

First of all I would like to thank my supervisors for the useful discussions and their critical remarks. I also like to thank my employer, the Dutch Ministry of Defence. They supported this work through facilitating in time and college funds.

Note that all statements of fact, opinion or conclusions contained herein are those of the author and should not be construed as representing the official views or policies of the Dutch Government.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Context

The idea of building a quantum computer already exists for 35 years [2]. Recently companies like IBM[1] and Rigetti[2] made quantum computing on real quantum computers available in the cloud. Many organizations look with great eager to this new type of computers, because these are supposed to be of great value for research in the field of medicine, material science and energy [3]. Quantum computers also promise to solve computational problems in the research area's of transport, logistics and artificial intelligence [4]. However the opportunities promised by the quantum computer also come at a price.

Several sources, e.g. NIST [5], ETSI [6] and the Dutch NCSC [7] report that the quantum computer is a threat to cyber security. To be more precise: some quantum algorithms pose a threat to some algorithms we use to protect our data. The quantum algorithm proposed by P. Shor poses a threat to cryptographic algorithms based on the (abelian) hidden subgroup problem, such as the mathematical problem of prime factoring as used in RSA [8]. Shor's algorithm is supposed to solve the factoring problem exponentially faster than known classical algorithms. This would result in breaking RSA, a widely used asymmetric cryptographic algorithm used for key agreement and transport but also for digital signature generation and verification [9].

Another quantum algorithm that poses a threat is Grover's algorithm. This quantum search algorithm speeds up brute-force attacks on symmetrical key algorithms, such as

[1] IBM Q Quantum experience (https://quantumexperience.ng.bluemix.net/qx/experience, consulted on 24-10-2017).

[2] Rigetti Forest 1.0 (https://medium.com/rigetti/introducing-forest-f2c806537c6d, consulted on 24-10-2017) The platform provides a quantum simulator and a quantum chip.

AES. This quantum algorithm also affects secure hash functions, such as SHA-2 and SHA-3. However the speed up Grovers algorithm promises, which is quadratic is, is less compared to the speed up of Shor's algorithm, which is exponential. Therefore the threat posed by Grover's algorithm is considered to be less significant [8]. The focus of this thesis is therefore on Shor's algorithm.

The in this thesis investigated risk event is the event that Shor's algorithm can be run on an available quantum computer and will break RSA with a 2048 bit key (RSA-2048).

Getting insight in the impact and the probability of an identified risk event is part of the risk analysis phase of a risk assessment. The impact of the defined risk event depends strongly on the context, consequences and current repressive countermeasures. All of these are specific to the organization involved and the responsibility of the person performing the risk assessment, the cyber security specialist.

The probability or likelihood that there is a quantum computer that is able to run Shors algorithm and break RSA-2048, is a general "feature" from the cyber security specialists point of view. Investigating the likelihood of the risk event is the focus of this thesis.

Some work has been done on this topic, including calculating the cost of running Grover's algorithm for SHA-256 and SHA3-256 [8]. Also a cost estimation for running Shor's algorithm to break a 2000-bit number has been reported [10]. These cost estimations can be understood as requirements for a quantum computer and are one of the elements influencing the likelihood of the identified risk event. Another part of the likelihood is assessing if these requirements can be met and, if not, what it takes to close the gap between the requirements and the available resources.

Currently the views on how to deal with the quantum threat differ. One view is that there is plenty of time left for organizations and the other view is to start now with quantum risk assessments [11].

The second objective of this thesis is to determine how to deal with the quantum threat, based on the findings of understanding the requirements to run Shor's algorithm and break RSA-2048.

1.2 Research questions

The main objective of this thesis is to understand the requirements to run Shor's algorithm and break RSA-2048. And how to use this information to protect against the quantum threat.

Sub question 1: What is a quantum computer and which related research is done regarding the cyber security threat quantum computers pose?

Sub question 2: What is Shor's algorithm and what are the building blocks on the logical layer?

Sub question 3: How can these logical building blocks be implemented on a practical quantum computer and what are the costs for running Shor's algorithm for a significant key size?

Sub question 4: What is the gap between the physical resources of current quantum computers and the cost requirements of running Shor's algorithm for a significant key size?

Sub question 5: Which implementation factors reduce the gap?

The second objective of this thesis is to answer the question: How to deal with the identified quantum threat?

1.3 Methodology

1.3.1 Methodology for answering the research questions related to quantum computing and technology

The quantum computing and quantum technology related research in this thesis follows a methodology derived from methods applied in [8, 10, 12]. Additionally the most likely types of physical implementations and their physical resources are compared to the cost required to run Shor's algorithm to determine the gap between the supply side, represented by the physical layer and the demand side, represented by three layers. These three layers are the application layer, the logical layer and the fault-tolerant layer. The used method is shown in Figure 1.1.

The first step is to analyze Shor's algorithm and investigate possible circuits implementing the algorithm. Next is investigated how a translation can be made to practical implementations of logical circuits, by applying fault-tolerant architectures and determining the required physical resources. The last step is to determine which qubit technologies are able to meet the required physical resources and what is currently the state of these qubit technologies. This information is used to determine the gap between the demand side and the supply side.

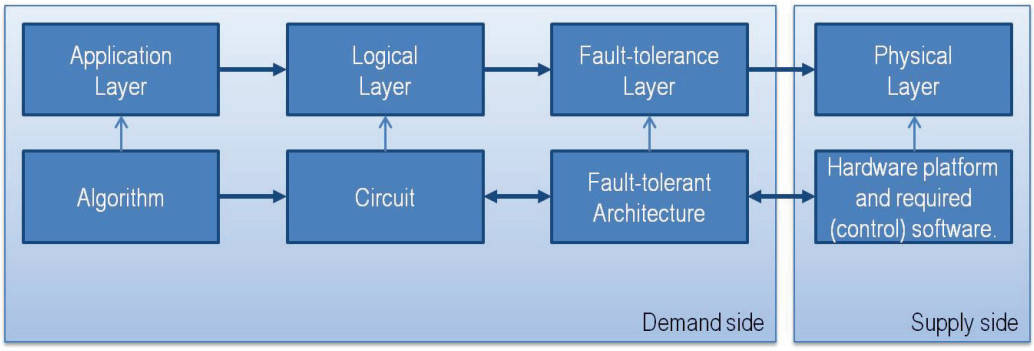


FIGURE 1.1: Method to determine the requirements for running Shor's algorithm and the gap between the demand side and the supply side.

Two experts from different quantum research niches reviewed the assumptions made in this thesis. These experts are R. Versluis and C. Schaffner. Dr. Christian Schaffner[3] is professor at the institute for logic, language and computation (ILLC) at University of Amsterdam and works as a researcher at QuSoft. Ir.Richard Versluis[4] is a principal scientist/systems engineer at TNO and lead scientist for TNO at QuTech.

In this research only public sources are used as information source. The focus is on literature written by researchers in the quantum domain and the predictions made by quantum experts.

To obtain sufficient knowledge about Shor's algorithm, information is used from lecture notes and other public sources used to teach university students Quantum Computing or Quantum Information Processing. As with all algorithms, Shor's algorithm needs to be made "mechanically" in order to run it on a computer. For this circuits are used. This defines the next step, to look at different logical implementations of Shor's algorithm using logical quantum circuits. Most information on this topic is obtained from review papers summarizing differences between quantum circuits implementing Shor's algorithm.

To answer the third research question a literature study is performed to investigate the practical building blocks and requirements needed to run the quantum circuit. We have chosen to focus on the most mature technologies, these are obtained from recent scientific review papers. The fourth research question is answered after a short inventory of the main physical implementations. To answer the fifth research question the most costly building blocks on the demand side are selected and on the supply side a selection is based on the requirements from the demand side.

[3]https://www.cwi.nl/people/2134
[4]https://qutech.nl/person/richard-versluis/

1.3.2 Methodology for answering the research question how to deal with the quantum threat

To answer the last research question, first a reflection on the likelihood of the risks related to the quantum threat was given based on the findings from the previous research questions.

Also a reflection on the impact of the quantum threat was given. Additional to what was found in the literature on the technical impact of the quantum threat an observation was made that the crypto-systems that are vulnerable to the quantum threat are frequently applied as risk controls to mitigate a risk to an acceptable risk level. This observation is used to develop a method for translating the technical impact of the quantum threat to the impact on the strategic risks of an organization.

The information published by ETSI is used to formulate the impact of the quantum threat on society. A description of the actors involved in the response to the quantum threat regarding cyber security is given based on information from the Dutch second national cyber security strategy (NCSS 2).

The reflections on impact and likelihood are used to reflect on an existing quantum risk assessment methodology by Mosca & Mulholland and two other options, delaying an action plan and mitigation without risk-assessment. A new method for handling the quantum threat is proposed using the findings from the reflection on likelihood and impact.

The proposed monitoring framework uses the findings from research question 5 and the information from standardization institutes ETSI and NIST to determine the topics to monitor. Information from the NCCS2 is used the determine the actors involved for providing the obtained information from monitoring to the relevant parties.

1.3.3 Limitations of the research and intended audience

The literature review to investigate the resources required to run Shor's algorithm and break RSA-2048 was limited to public sources. A selection from the large amount of available and suitable material, has been made consulting experts.

The investigation about the supply side was limited to the quantum chip, the hardware platform which contains the physical qubits. The impact of the other elements needed for the physical realization of a quantum computer, see Chapter 2 were not taken into account when determining the gap between the supply side and the demand side.

Another limitation was to use RSA-2048 to determine the physical requirements. There are other crypto-systems vulnerable for the quantum threat, which will lead to other physical requirements and other gaps. To make a more complete analysis of the quantum threat, these other algorithms and their physical requirement should be investigated.

This thesis has been written for people working in the cyber security field with a background in risk management and who are familiar with risk-management standards like NEN-ISO/IEC 31010. Only a general background in quantum mechanics, computer science, information science or electronics is needed to read the complete thesis. Some basic math is required from the reader as well as knowledge about RSA. Readers with no background in quantum mechanics, computer science, information science or electronics can skip Chapters 3 and 4 to get an overview of the results and the reflection on how to deal with the quantum threat.

The thesis uses a high abstraction level, more details on topics can be found in the relevant literature used in each chapter. A more detailed mathematical and physical description of quantum mechanics can be found for example in [2, 13].

1.4 Thesis structure

The research questions are answered using the following thesis structure.

Sub question	**Chapter answering the sub question**
1: What is a quantum computer and which related research is done regarding the cyber security threat quantum computers pose?	Setting the scene
2: What is Shor's algorithm and what are the building blocks on the logical layer?	Shor's algorithm and logical building blocks for implementation
3: How can these logical building blocks be implemented on a practical quantum computer and what are the costs of running Shor's algorithm for a significant key size?	Fault-tolerant implementations and hardware options
4: What is the gap between the physical resources of current quantum computers and the cost requirements of running Shor's algorithm for a significant key size?	Fault-tolerant implementations and hardware options
5.Which implementation factors reduce the gap?	Fault-tolerant implementations and hardware options
How to deal with the identified quantum threat?	Reflection on the quantum threat
Conclusions and recommendations for further research	Conclusions and further research

Table 1.1: Overview of research questions and chapters.

Chapter 2

Setting the scene

Most cyber security experts are familiar with threats in the classical domain[1]. For example, vulnerabilities in IT systems or industrial control systems, vulnerabilities resulting from the human component in IT, such as attackers using social engineering to get access to information or IT systems and vulnerabilities resulting from procedures, such as inadequate patch management.

To understand the cyber security threat a quantum computer poses, a brief introduction in the quantum field is necessary. This chapter starts with a brief introduction about what a quantum computer is and closes with more background information on the threat quantum computers pose to cyber security. The threat quantum computers pose is often referred to as the quantum threat. In this thesis both are used.

2.1 What is a quantum computer?

The definition of a quantum computer strongly depends on the scientific field and the background of the audience. For the purpose of this thesis a general definition is chosen:

A quantum computer is a computing device that exploits quantum-mechanical phenomena, such as superposition, entanglement and interference.

This definition is derived from a similar definition as published in [14]. The next two sections provide more information on the quantum mechanical phenomena and the computing device.

[1] People working in the in the scientific fields related to quantum mechanics, quantum computing, quantum information theory etc, use the term classical for the world as we know it or everything that is not using the unique quantum mechanical features explicitly. Note that the term conventional is also used.

2.1.1 Quantum-mechanical phenomena

The computational power of a quantum computer is based on three phenomena: quantum parallelism, entanglement and quantum interference. These phenomena provide quantum algorithms the computational power to solve certain mathematical problems more efficiently than classical computers solve these problems.

Quantum parallelism uses the quantum feature of superposition to apply a function[2] to all possible input values simultaneously. Superposition allows quantum systems to be in many different states at the same time [13]. Quantum systems are used to store information. The most simple quantum system is a qubit. In press releases, superposition is often translated as: a qubit can be 1 and 0 at the same time.

"Entanglement arises when two or more quantum systems exist in a superposition of correlated states."[15]. If for example two qubits were entangled and separated physical from each other[3], then the operations performed on the first qubit would affect the second qubit instantaneously independent of the distance between them. These are called non-local effects and refer to what Einstein called "spooky action at a distance" [13].

The effects of interference on an algorithm can be best compared with interference patterns between light or sound waves [13]. Depending on the applied interference, some quantum states cancel out and some are amplified. When measuring a quantum state to obtain the output of an algorithm, this amplification means that the probability of obtaining a measurement result (output) from that amplified quantum state increases and the probability of obtaining a measurement state from a less amplified state decreases.

2.1.2 Computing device

Here we formulate a computing device as a system with an input, some computation (evolution) and an output, see Figure 2.1. For example: Shor's algorithm has as input, integer $N = pq$, where p and q are large prime numbers. The goal is to compute the prime factors p and q as desired[4] output. Quantum computation can be defined as *"a sequence of unitary transformations, affecting simultaneously each element of the superposition, generating a massive parallel data processing albeit within one piece of quantum hardware"* [16].

[2] Algorithms apply one of more mathematical function(s).

[3] For example: one qubit stays in The Hague (city in Europe) and the other is transported to a city in Australia.

[4] Note that it can be verified on a classical computer that the output is of the desired form ($N = pq$).

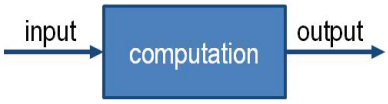


FIGURE 2.1: Computing device using an input to produce an output.

A quantum computer needs a quantum state as input. The most simple quantum state is a qubit. The state of a qubit can be visualized as a point on a unit three-dimensional sphere,the Bloch sphere, as visualized in Figure 2.2.

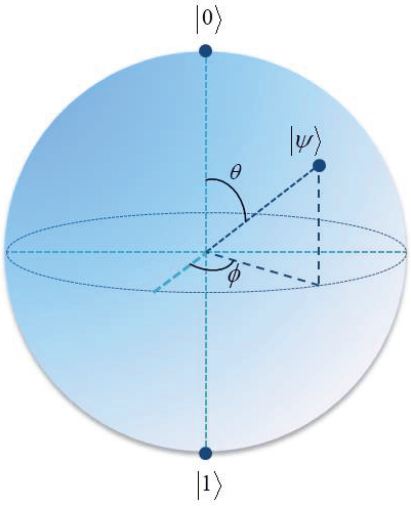


FIGURE 2.2: Bloch sphere representation including three different qubit states $|0\rangle$, $|1\rangle$, $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, where α and β are complex numbers. Because $|\alpha|^2 + |\beta|^2 = 1$, $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle = e^{i\gamma}\left(cos\frac{\theta}{2}|0\rangle + e^{i\phi}sin\frac{\theta}{2}|1\rangle\right)$, where θ, ϕ and γ are real numbers. Since $e^{i\gamma}$ has no observable effects, this term is omitted, resulting in $|\psi\rangle = cos\frac{\theta}{2}|0\rangle + e^{i\phi}sin\frac{\theta}{2}|1\rangle$ [2].

After initialization of the quantum state (the input), the quantum state is evolved to an output state using a quantum circuit with quantum gates. During the execution of the algorithm the quantum states may suffer from an effect called decoherence.

Decoherence is a kind of noise that has no classical analog. A simplified model of decoherence is described by two parameters acting in parallel, phase damping and amplitude damping. Phase damping describes the loss of quantum information without loss of energy [2]. It is called phase damping, because it randomly changes the phase of a quantum system reducing the coherence between the superposed $|0\rangle$ and $|1\rangle$ states and by this leaking information. Phase randomization has timescale T_2 [2, 17]. Phase damping is visualized in Figure 2.3. Phase damping affects the xy-plane of the Bloch sphere, $x_1 \neq x_2$ and $y_1 \neq y_2$.

Amplitude damping refers to the effects due to energy loss in a quantum system [2]. For example if a qubit in superposition, $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, where $|\alpha|^2 + |\beta|^2 = 1$, loses energy the amplitude factor β is reduced, while the the amplitude factor α increases,

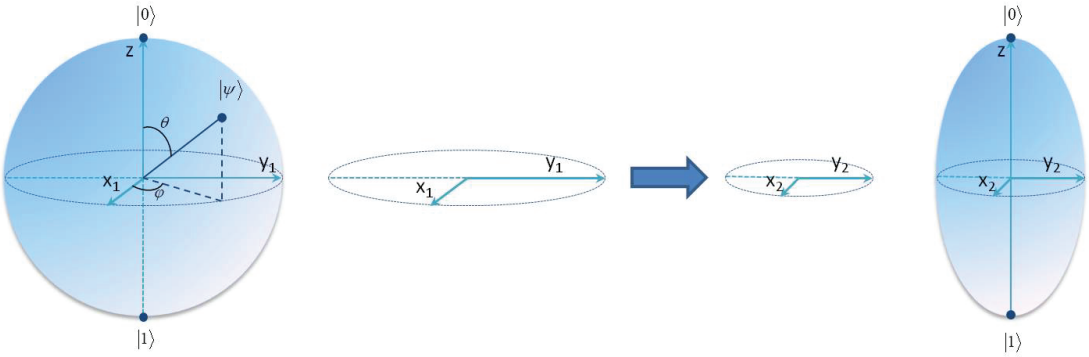


FIGURE 2.3: Visualization of phase damping. On the left the Bloch sphere without phase damping effects, the xy-plane forms a perfect unit circle. On the right the Bloch sphere with phase damping effects, the xy-plane is not forming a perfect unit circle.

driving the qubit to its ground state. Amplitude damping, or relaxation, has timescale T_1. In principle the physical qubit relaxation time T_1 and the physical de-phasing time T_2 determine the coherence time.

When an algorithm is executed errors occur, due to decoherence and other effects like control errors and measurement errors. To execute an algorithm successfully all required computational steps should be completed with a low error rate per computational step. The error rate limits the practical number of computational steps that can be executed by a quantum computer. However by encoding quantum information in multiple physical qubits, the available execution time can be increased to exceed significantly the physical coherence time of the physical qubits. To encode quantum information in multiple physical qubit states quantum error correction (QEC) codes are used.

Example: Quantum Error Correction Code

Quantum systems can protect a single quantum state, e.g. $|1\rangle$, by encoding this quantum state using three quantum states, creating redundancy in the information. The resulting protected quantum system is called a logical quantum state and is stored in three qubits.

$$|1> \rightarrow |1\rangle_L \equiv |111\rangle \text{ , where } L \text{ stands for logical state.}$$

The logical quantum state $|1\rangle_L$ representing a bit of information with value one is encoded in three physical qubits $|111\rangle$.

To perform a computation on the logical qubit, computations need to be applied to all three physical qubits, more on this in Chapter 4. If at most one of the three qubits obtains an error during computation, then the QEC code can recover using the majority vote.

There are many QEC codes and they all have their own properties. QEC codes use the notation $[n, k, d]$, where n represents the number of physical qubits, k denotes the number of logical qubits, and d is the distance of the code [18]. The number of errors[5] a code can correct depends on the code distance d.

Using QECs to protect the information stored in the quantum system as it dynamically undergoes computation is called fault-tolerant quantum computation [2]. By increasing the code distance d, arbitrary low logical error rates can be achieved. In an architecture realizing fault-tolerance different QEC codes can be used to achieve arbitrarily good quantum computation. Which QEC codes are optimal and which corresponding values of d, depend on the the type of error model, the quality of the physical qubits and architectural considerations, such as the connectivity between qubits. A fault-tolerant architecture can be realized [19]. It will however come at a price, which is reflected in the requirements on the physical resources, see Chapter 4 for more details.

In the end, when the input quantum state has been evolved to the final quantum state, a measurement needs to be done to obtain the results from the computations. The effects of superposition, a quantum state being in multiple states at the same time, ends when measuring this state. During measurement, using a computational basis ($|0\rangle$ and $|1\rangle$), the quantum state is forced into one of many quantum states. For example, a qubit in superposition ($|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$) is forced into a 1 or a 0, it cannot be both. The probability[6] of measuring the desired output should be high to increase the success rate of running the algorithm.

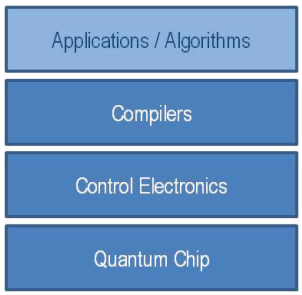


FIGURE 2.4: General visualization of layers needed for the physical realization of a quantum computer, derived from [20].

To realize a computing device with a fault-tolerant architecture and the capability to initialize quantum states, to perform computational steps and to measure the final quantum states, physical building blocks are required. A physical realization of the quantum computer consists not only of a quantum chip, but also includes the complementary hardware and software. This is represented by the bottom three layers of Figure 2.4.

[5] *"The distance between two code words states, d, defines the number of errors that can be corrected, t, as,*$t = \lfloor (d-1)/2 \rfloor$*"*[19].

[6] Measuring results in either 0 with probability $|\alpha|^2$, or 1 with probability $|\beta|^2$.

The bottom layer, the quantum chip, holds the physical qubits which are structured to enable a fault-tolerant architecture. These physical qubits need to be controlled, to enable physical qubit operations including measurements for error detection, this is represented in the control electronics layer. The compiler layer enables optimization, error corrections and logical operations [20]. Chapter 4 will provide information about two types of physical qubit implementations, also referred to as qubit technologies and quantum chips.

2.2 The quantum threat and cyber security

In 2016 NIST published a table summarizing the threat quantum computers pose, see Table 2.1. NIST explains that many communication protocols use three core cryptographic functionalities: public-key encryption, digital signatures, and key exchange. Diffie-Hellman key exchange, the RSA cryptosystem, and elliptic-curve cryptosystems are mostly implemented to fulfill these functionalities [5]. However their security depends on the difficulty of solving problems such as integer factorization or the discrete-log problem over various groups.

In [21] P. Shor showed that both problems can be efficiently solved on a quantum computer, and -as the NIST publication puts it- *"thereby rendering all public-key cryptosystems based on such assumptions impotent"* [5]. The quadratic speed up of Grover's search algorithm has a less significant impact. The NIST publication explains that cryptographic systems should not be considered obsolete, but there will be a need for larger key sizes even for symmetric-key algorithms [5].

Cryptographic Algorithm	**Type**	**Purpose**	**Impact from large-scale quantum computer**
AES	Symmetric key	Encryption	Larger key sizes needed
SHA-2, SHA-3	————	Hash functions	Larger output needed
RSA	Public key	Signatures, key establishment	No longer secure
ECDSA, ECDH (Elliptic Curve Cryptography)	Public key	Signatures, key exchange	No longer secure
DSA (Finite Field Cryptography)	Public key	Signatures, key exchange	No longer secure

TABLE 2.1: Derived from NIST Table - Impact of Quantum Computing on Common Cryptographic Algorithms [5].

Similar work is provided by other institutes, like ETSI. They provide a comparison of security levels, both conventional and quantum, for some popular ciphers including RSA, ECC and AES [6].

The NIST publication states that only a "large-scale quantum computer" will impact the discussed cryptographic algorithms [5]. About the likelihood that there is or will be a large scale quantum computer, the NIST publication only refers to predictions from experts. *"While in the past it was less clear that large quantum computers are a physical possibility, many scientists now believe it to be merely a significant engineering challenge. Some experts even predict that within the next 20 or so years, sufficiently large quantum computers will be built to break essentially all public-key schemes currently in use"* [5]. For this prediction NIST used the estimates made by expert M. Mosca. Section 2.2.1 will elaborate more on his work.

2.2.1 Related work

M. Mosca derived a simple model to determine when it is time to prepare for the quantum threat, Mosca's "x, y, z" quantum risk model. This model includes the duration that information should be kept secure (x), the time it takes to migrate to a quantum-safe solution (y) and an estimate on when identified threat actors have access to quantum technology (z) [22], see also Section 5.3.2.

To determine z, when threat actors get access to a large-scale quantum computer, M. Mosca makes estimations *"one in seven chance that some of the fundamental public-key cryptography tools upon which we rely today will be broken by 2026 and a 50% chance by 2031."* [23, 24]. These estimates are based on several key values, in Mosca's paper three of these values are described [24].

The first value is: *"When will we reach the design of a fault-tolerant scalable qubit?"* A target date of the year 2021 was given in 2015 by an IARPA[7] announcement for proposals to build: *"a logical qubit from a number of imperfect physical qubits by combining high-fidelity multi-qubit operations with extensible integration"* [24] [8].

The second value is: *"How many physical qubits will we need to break RSA-2048? ... Current estimates range from tens of millions to a billion physical qubits."* These estimates depend on a lot of factors. Mosca names a subset: the efficiency of fault-tolerant error-correcting codes, the physical error models and error rates of the physical quantum computers, optimizations in the quantum factoring algorithms, and the efficiency of the synthesis of factoring algorithms into fault-tolerant gates [24].

[7] Intelligence Advanced Research Projects Activity (IARPA) US Government.
[8] Mosca's paper refers to http://www.iarpa.gov/index.php/research-programs/logiq

The last value is: *"How long will it take to scale the scalable design to the size sufficient to break RSA-2048?"* Mosca notes that the rate of scaling depends on the availability of tools, some of which are already available and some of which are being developed [24].

For the second value given by Mosca, a kind of structured approach is provided in Appendix M of a paper by Fowler et. al. [10]. In this appendix they estimate the amount of physical qubits for running Shor's algorithm to factor $N = pq$, with N of size 2000 bits, based on a surface-code implementation for fault-tolerance. Similar work is done by Amy et. al. for running Grover's algorithm to break SHA-256 and SHA3-256 in [8]. Both use a similar method: choose the algorithm you wish to run, choose a circuit implementing the algorithm (or design/optimize a circuit), choose a fault-tolerant implementation and determine the required number of physical qubits and other physical resources.

These physical resources need to be available on a quantum computer in order to run the algorithm using the circuit. There are many different options for implementing the required physical building blocks. Two papers, one using spin qubits and the other using Josephson charge qubits, have implemented Shor's algorithm factoring, integers $N = 15$ and $N = 21$ [25, 26]. These papers omit the fault-tolerant implementation and directly implement the circuit on the physical layer. They do however provide a bridge between two separate research fields, the research field studying quantum algorithm and the field studying the physical realization of quantum building blocks [26].

Not only the realization of the physical building blocks is important but also the way these building blocks work together. In a paper by R. van Meter et al. is stated that the architecture used for the realization of a quantum computer *"can make the difference between an interesting proof of concept device and an immediate threat to all RSA encryption"*[27]. He makes this concrete by comparing the best known classical threat to RSA, the number field sieve, with Shor's algorithm implemented on different architectures running with different clock speeds. For a N of size 1000 bits the best known classical algorithm takes more than a thousand years to factor N. The time needed to factor a N of this size using Shor's algorithm on different architectures ranges from seconds to more than a thousand years [1].

This chapter provided a short overview on what a quantum computer is: a computing device that exploits quantum-mechanical phenomena, such as superposition, entanglement and interference. We also learned that a fault-tolerant architecture is needed to cope with control and measurement errors and errors caused by decoherence. However fault-tolerance will come at a price, which will be explained in Chapter 4. The next chapter will provide an overview of Shor's algorithm and relevant quantum circuits implementing this algorithm.

Chapter 3

Shor's algorithm and logical building blocks for implementation

This chapter investigates the building blocks needed to run Shor's algorithm efficiently. First a description of Shor's algorithm is given, followed by an introduction on quantum circuits. These quantum circuits are used to implement algorithms. An overview is provided of quantum circuits relevant for Shor's algorithm and this chapter closes with a reflection on the general building blocks needed to run Shor's algorithm for a 2048-bit integer $N = pq$.

3.1 Shor's Algorithm

In 1994 Peter Shor showed that two important problems, for which we do not know any efficient classical solution, could be solved efficiently on a quantum computer. He gave a quantum solution for the problem of finding the prime factors of an integer and a solution for the so-called discrete-logarithm problem [21].

In his paper, P. Shor shows that the problem could be solved in polynomial time by dividing it in four steps [21]. To achieve this speed up only one of these steps, Step II, needs to be executed on a quantum computer. The other three steps are executed on a classical computer [15].

Shor's Algorithm

Let $N = pq$ for two large prime factors p and q. In order to find p and q, follow the steps below.

- STEP I: Choose x such that $1 < x \leq N - 1$.

  Compute the Greatest Common Divisor (GCD) of x and N to make sure that they are relative prime ($GCD(x, N) = 1$).

  Note that if $GCD(x, N) \neq 1$, then $GCD(x, N) = p$ or q and we can stop the algorithm.

- STEP II: Solve the discrete-logarithm problem for a given x and N, i.e. find the smallest hidden period r, such that $x^r \equiv 1 \mod N$.

- STEP III: Check if r is even. If r is odd, then restart at STEP I. If r is even, then derive:

$$\begin{aligned} x^r &\equiv 1 \mod N \\ x^r - 1 &= 0 \mod N \\ x^r - 1 &= cN \qquad \text{(where } c \text{ is an integer.)} \\ (x^{r/2} - 1)(x^{r/2} + 1) &= cpq \end{aligned}$$

- STEP IV: Calculate $p = GCD((x^{r/2} - 1), N)$ and $q = GCD((x^{r/2} + 1), N)$.

Note for STEP I that the GCD can be calculated using Euclid's algorithm:
$GCD(x, N) : r_1 = N \mod x$, where $0 \leq r_1 \leq x - 1$;$r_2 = x \mod r_1$, where $0 \leq r_2 \leq r_1$; $r_3 = r_1 \mod r_2$, where $0 \leq r_2 \leq r_1$; ...; $r_n = 1; gcd = r_{n-1}$ [15].

Note for STEP II the periodicity results from repeated multiplication with x. The sequence: $1 = x^1 \mod N, x^2 \mod N, x^3 \mod N, \ldots$, will start to cycle after a while: there is at least an $0 < r \leq N - 1$, for which holds $x^r = 1 \mod N$, where r is called the period of the sequence [13]. For an example, see Appendix A.

Note for STEP III it can be proven that r is even with a probability ≥ 0.5 that and $(x^{r/2} - 1)$ and $(x^{r/2} + 1)$ are no multiples of N. The proof, using basic number theory, can be found in [13] on page 29.

STEPs I, III, IV can all be run on conventional computers. The challenge arises at STEP II, the quantum step of Shor's algorithm. This step will be further explained after a short introduction to quantum circuits.

3.2 A brief introduction to quantum circuits

To understand how STEP II can be implemented on a quantum computer, quantum circuits should be investigated.

Quantum circuits are made of wires and gates that together implement an algorithm. An algorithm is a precise recipe for performing a computational task, e.g. prime factorization [2]. A gate maps an input quantum state to an output quantum state. The wires represent the quantum states and show the connections between input and output of the gates. This is summarized in Figure 3.1.

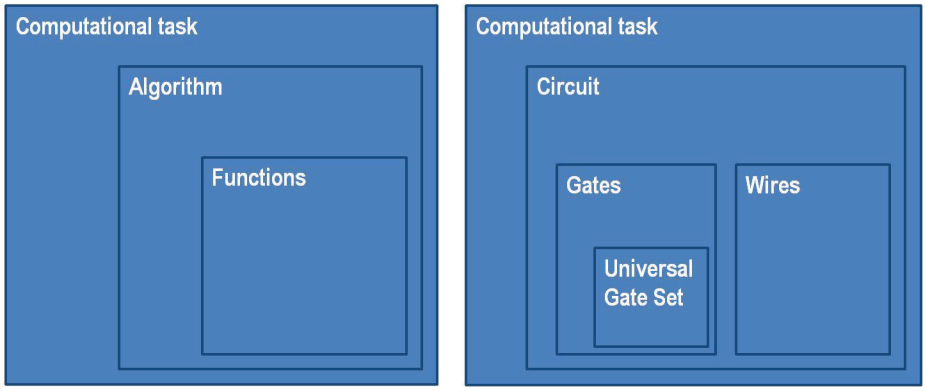


FIGURE 3.1: The different elements of implementing a computational task.

Figure 3.2 shows an example of a simple quantum circuit. Any quantum operation (gate) can be represented by an $M \times M$ complex-valued matrix U, which is an unitary matrix, meaning that its inverse U^{-1} equals its conjugate transpose U^*. Any quantum operation is by definition reversible.

Example: Simple quantum circuit

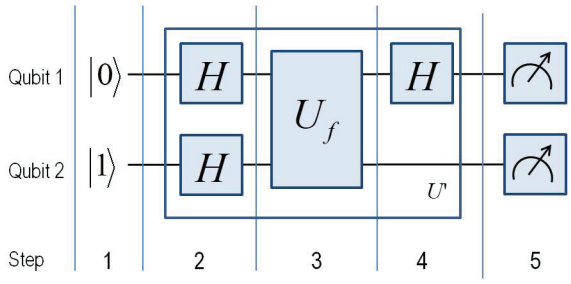


FIGURE 3.2: Example of a simple quantum circuit.

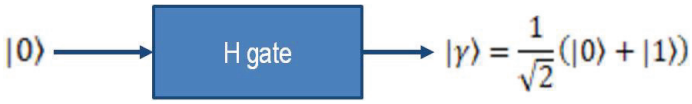
A simple quantum circuit consisting of the following operations: 1. qubit initialization, 2. single-qubit gates, 3. two-qubit gate, 4. single-qubit gate, 5. a measurement.

A quantum state (the state of the wire) $|\phi\rangle$ is represented as a M-dimensional vector $(\mu_1, \ldots, \mu_M)^T$. If a unitary operator transforms this quantum state to an output state $|\psi\rangle$, which is represented by a M-dimensional vector $(\lambda_1, \ldots, \lambda_M)^T$, then this will be denoted as $|\psi\rangle = U|\phi\rangle$ or:

$$\begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{pmatrix} = U \begin{pmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_n \end{pmatrix}.$$

Depending on the complexity of the quantum operation the quantum gate may be built from other gates implementing parts of the quantum operator. There are many different gates. A set of gates is called universal if all other unitary transformations can be built[1] from that set [13], see Figure 3.1. Figure 3.3 shows an example of an important single-qubit gate, the Hadamard gate, which maps the input state $|0\rangle$ to an output state $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, a superposition.

Example: Hadamard gate



Calculation: $|\gamma\rangle = H|0\rangle$, with $H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$

$$= \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}\begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

$$= \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 0 \end{pmatrix} + \frac{1}{\sqrt{2}}\begin{pmatrix} 0 \\ 1 \end{pmatrix} = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

FIGURE 3.3: Example of a gate operation: the input $|0\rangle$ is multiplied by the Hadamard transform implemented by a Hadamard gate. This produces an output in superposition: $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$.

A different type of quantum operator is the measurement operator, which is non-reversible. When measuring in the computational basis, the quantum state $|\phi\rangle = (\mu_1, \ldots, \mu_M)^T$, is forced into the classical state $|j\rangle$ with probability $|\mu_j|^2$, this is the squared norm[2] of the amplitude μ_j [13]. There are other types of measurements that do not force a quantum state into a classical state, this type of measurement is needed for a fault-tolerant implementation of Shor's algorithm and will be addressed in Chapter 4.

[1] The set of all single-qubit operations together with the two-qubit CNOT gate is universal [13]. Other sets approximate any other unitary arbitrarily well using circuits of only these gates. It has been proven in the Solovay-Kitaev theorem that this approximation can be done efficiently [13].

[2] Amplitude μ_j is a complex number $\mu_j = a + ib$, where a and $b \in \mathcal{R}$ and $i^2 = -1$, the squared norm: $|\mu_j|^2 = \sqrt{a^2 + b^2}$.

In this short introduction to quantum circuits we have seen that algorithms are represented by quantum circuits. These circuits are built using quantum gates and wires. Quantum gates implement unitary transforms which makes the gates reversible. In the next section, Step II of Shor's algorithm is discussed and visualized using a general circuit implementation.

3.3 The quantum step in Shor's algorithm

STEP II, finding the smallest period r of $x \mod N$, is called order finding, because in the discrete-logarithm form factoring has a periodic structure [15]. The process of finding the order r can be summarized in six main steps [15, 21, 25, 26].

1. Initialize quantum registers.
2. Generate a superposition on the qubits in the first register.
3. Compute the modular exponentiation: $f(x) = a^x \mod N$, for a given N, a and x and use the second register to store the values of $f(x)$.
4. Apply the Quantum Fourier Transform (QFT) to the first register.
5. Measure the first register to get an indication of the period r.
6. Use a classical computer to perform post-processing to obtain the period r.

Figure 3.4 shows a general implementation of Step II, the steps are executed from the left to the right. The lines represent the wires and the blue boxes the gates implementing the quantum operations. The last box on the right (CP) is a classical post-processing step, and not a quantum gate. The steps are briefly explained using various sources [15, 21, 25, 26]. The description of the modular exponentiation, Step 3, also uses the V.Meter's thesis [1] as a source.

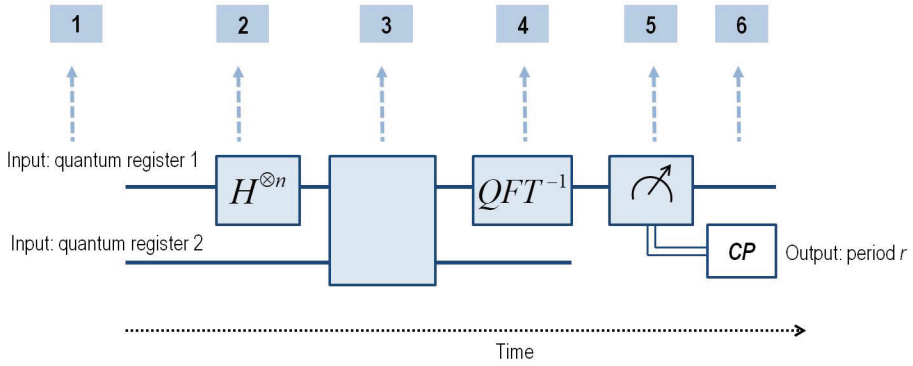


FIGURE 3.4: Example of a gate implementation of Step II of Shor's algorithm, derived from [25, 26]. CP refers to Classical Post-Processing.

3.3.1 Initialization

To start the procedure of finding primes p and q, two quantum registers are prepared. The first quantum register contains $n = 2\lceil log_2 N \rceil$ qubits in state $|0\rangle$, where $N = pq$. The second quantum register contains $m = \lceil log_2 N \rceil$ qubits in state $|1\rangle$. The first quantum register is used to store the values of x, the second quantum register will be used to store the values of $f(x) = a^x \mod N$.

3.3.2 Superposition

The next step, Item 2 of Figure 3.4, puts the quantum states of the first register in superposition by applying the Hadamard gate to each qubit individually ($H^{\otimes n}$). This results in the state: $\frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle|0\rangle$. Note this is similar to the example in Section 3.2, only now for n qubits.

3.3.3 Modular Exponentiation and entanglement

Item 3 of Figure 3.4 contains two steps, modular exponentiation and entanglement. The modular exponentiation problem is: Compute $f(x) = a^x \mod N$, for a given N, a and x. The result, the values of $f(x)$, are stored in the second register. Computing the modular-exponentiation problem is considered the most resource-intensive step of the quantum part of Shor's algorithm, as it consumes the most time and space.

Modular exponentiation is realized using modular multiplication building blocks, which on their turn are realized using blocks that perform modular addition. These last blocks are built from blocks performing addition and comparison, see Figure 3.5. The construction of the modular multiplication using these two building blocks can be compared to a classical variant of writing an exponent as a set of multiplications and a multiplication as a sum. Mathematically: $\alpha^\beta = \underbrace{\alpha \cdot \alpha \cdot \cdots \cdot \alpha}_{\beta \text{ times}}$ and $\nu \cdot \mu = \underbrace{\nu + \nu + \cdots + \nu}_{\mu \text{ times}}$.

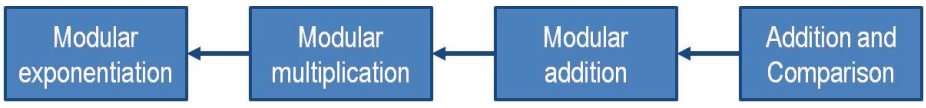


FIGURE 3.5: Building blocks of modular exponentiation, derived from [1].

The quantum building blocks realizing the add-functionality are referred to as *quantum adders* and the quantum building blocks realizing the multiplication-functionality are referred to as *quantum multipliers*. There are many different implementations of these building blocks, as we will see in Section 3.4.

Entanglement

The next step is to use a unitary operator U_f to entangle the input state x with the corresponding value of $f(x)$, this results in the state: $\frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle|a^x \mod N\rangle$. This state is periodic with (hidden) period r.

The periodicity results from repeated multiplication with x. The sequence: $1 = x^1 \mod N, x^2 \mod N, x^3 \mod N, \ldots$, will start to cycle after a while: there is at least an $0 < r \leq N - 1$, for which holds $x^r = 1 \mod N$, where r is called the period of the sequence [13]. For an example, see Appendix A.

3.3.4 Quantum Fourier Transform

Using the Quantum Fourier Transform as a quantum operator, realizes the quantum-mechanical phenomena of interference. Applying the inverse Quantum Fourier Transform (QFT^{-1}) to the first register reveals the periodic property of the created state. This results in: $\frac{1}{2^n} \sum_{x,y=0}^{2^n-1} \exp(2\pi i x y/2^n)|y\rangle|a^x \mod N\rangle$. Positive interference, the amplification of the amplitudes, occurs when $y = c\frac{2^n}{r}$, for an integer c. For more detail see Appendix B.

3.3.5 Measurement

In this step the first register is measured. The positive interference caused a high probability of obtaining measurement result y for which holds that $\frac{y}{2^n} = \frac{c}{r}$, where y and 2^n are known and c and r not. Note that the exact value[3] of this probability depends on the value of r, the integer c and n, more information can be found in Shor's extended paper [21].

3.3.6 Classical post-processing

The last step, Item 6 of Figure 3.4, is to extract period r out of the obtained $\frac{y}{2^n} = \frac{c}{r}$. To find this period a technique called Continued Fraction Approximation is used. This technique can be implemented and run efficiently on a conventional computer, using classical computation. The mathematical steps can be found in e.g. [13].

These six steps are part of STEP II of Shor's algorithm, the process of finding the order r. We have seen that Shor proposed an probabilistic algorithm which factors $N = pq$, with large primes p and q, in four steps I-IV. The proposed algorithm does not only utilize the quantum-mechanical phenomena like superposition, entanglement and interference but also needs classical computation, that can be run efficiently on a conventional computer.

[3] The probability of obtaining the required output will be at least $\frac{1}{3r^2}$ if $|\{rc\}_{2^n}| \leq \frac{r}{2}$, with $|\{rc\}_{2^n}|$ the residue that is congruent to $rc \mod 2^n$, see Shor's extended paper [21].

3.4 Quantum circuits relevant for Shor's algorithm

To investigate the likelihood that there is a quantum computer that can run Shor's algorithm efficiently we focus on the most resource-intensive step of the quantum part of Shor's algorithm, the modular exponentiation. This part of the algorithm will dominate the physical requirements that are needed on a quantum computer [21].

To understand the physical requirements, first the logical requirements need to be investigated. Figure 3.6 shows the different layers for analyzing Shor's algorithm. The logical requirements depend on the optimization choices or restrictions applied while designing a modular exponentiation circuit. Providing insight in these design choices is the topic of the next section.

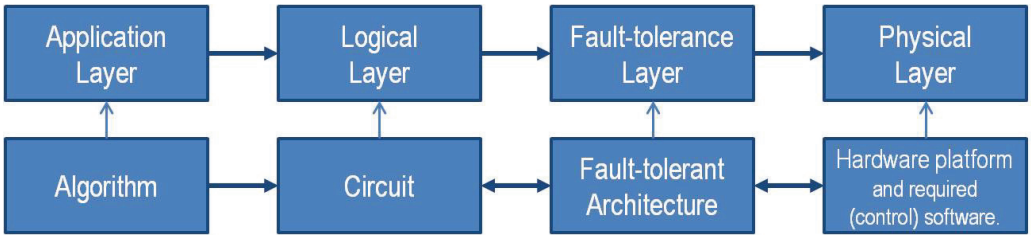


FIGURE 3.6: Different layers and the elements of these layers for analyzing the different resources required for running Shor's algorithm, based on [8].

3.4.1 Comparing quantum circuits

The quantum circuits discussed in this chapter, are executed in the logical layer, see Figure 3.6. These quantum circuits are compared using two parameters: The first parameter is the number of required logical computational qubits (wires) also referred to as the *circuit width*. The circuit width gives an indication for the space the quantum circuit requires. The second parameter is the logical circuit depth.

The *circuit depth* is defined as the number of sequential gates within a circuit [28]. This parameter is used as an indication for the duration of the algorithm. The actual execution time will depend on the implementation on the physical layer and on the different physical gate and measurement times[4]. However, to allow an early comparison of different circuits, the duration of all gates and measurements are assumed the same. This enables to express the duration or circuit depth in the number of gates.

Most of the time the values the circuit depth and width are expressed as indication of complexity. This is represented in orders, using the big O notation, or in a polynomial

[4]Note: on the physical layer is the measurement time longer compared to the gate execution times.

representation. Note the big O notation is also used for the comparison or analysis of algorithms in the field of classical computer science.

3.4.2 Overview of quantum circuits relevant for Shor's algorithm

In 2005 S. Devitt, A. Fowler and L. Hollenberg published an overview [28] of proposed circuit designs for modular exponentiation (Item 3 in Figure 3.4), their results are summarized in Table 3.1. The results provided are accurate to the leading order in m, where $m = log_2(N)$ and N is the integer of bit length m that needs to be factorized.

Circuit's optimization criteria	**Number of Qubits**	**Circuit Depth**	**Reference**
Conceptual simplicity	$\backsim 5m$	$O(m^3)$	"V.Vedral, A.Barenco, and A.Ekert, Quantum Networks for Elementary Arithmetic Operations, Phys. Rev. A 54, p. 147, 1996."
Running time	$O(m^2)$	$O(mlogm)$	"P. Gossett, Quantum carry save arithmetic, quant-ph/9808061, 1998."
Number of qubits	$\backsim 2m$ $(2m+3)$	$\backsim 32m^3$	"S. Beauregard, Circuit for Shors algorithm using 2n+3 qubits, Quantum Information and Computation 3, p. 175, 2003."
Time/Space - Trade-off I	$\backsim 50m$	$\backsim 2^{19}m^{1.2}$	"C.Zalka, Fast versions of Shors Quantum Factoring algorithm,quant-ph/9806084, 1998."
Time/Space - Trade-off II	$\backsim 5m$	$\backsim 3000m^2$	"C.Zalka, Fast versions of Shors Quantum Factoring algorithm, quant-ph/9806084, 1998."

TABLE 3.1: Overview of proposed circuit design's for Shor's algorithm.

Each quantum circuit is designed with some design objective or an optimization goal. For example the Beauregard circuit, third row in Table 3.1, is designed using the design criteria: *Minimize the number of (logical) qubits needed for factorization in polynomial time.*

The results above are just a snapshot in time. The field of quantum circuit design is continuously improving. For example in v. Meter's PhD thesis additional circuits are described. Most of these other circuits focus on reducing circuit depth for the adder gate building block [1]. When the the circuit depth and the number of gates is reduced, the number of computational qubits will increase, this is also referred to as the time/space - trade-off.

Selecting one of the circuits from Table 3.1 based on the best trade-off might be the best option. However the logical resources, number of computation qubits and the number of gates, for $m = 2048$ is significant and it is not yet clear how the logical resources can be translated to physical resources.

In a paper by A. Fowler et.al. [10] modular exponentiation circuits are compared using different parameters. This paper relates more to the fault-tolerant layer of Figure 3.6 and therefore considers the parameters: the number of computational logical qubits, the sequential number of Toffoli gates, and the total number of Toffoli gates. The reason for choosing these parameters is, that the physical size of the circuit scales with the ratio of the total number of Toffoli gates to the number of sequential Toffoli gates. To realize practical large-scale quantum computing, circuit designers should therefore minimize the number of these Toffoli gates used in their circuits.

3.5 Logical building blocks for running Shor's algorithm

For Shor's algorithm the building blocks can be roughly summarized as quantum state initialization, unitary transforms, measurement operations and classical processing. On the quantum logical layer the first three building blocks are translated to circuits using logical gates and logical qubits.

The required logical resources depend on design choices or optimization's applied when designing the circuits. When making these choices there is always a trade-off between the number of logical qubits and the number of logical sequential gates. This also known as the time-space trade-off, as the number of logical sequential gates give an indication for the duration of the computation and the logical qubits indicate an amount of space the circuit occupies.

For example Beauregards circuit minimizes the required logical qubits to implement Shor's algorithm. This circuit requires approximately $2m$ logical qubits and $32m^3$ logical sequential gates, where m is the bit length of the integer $N = pq$. Resulting in approximately 4099 logical qubits and approximately $275 \cdot 10^9$ sequential logical gates for $m = 2048$ bits.

Logical qubits and gates are capable of handling errors that occur. These errors occur due to decoherence, control errors, measurement errors etc. This capability of handling errors is realized by a fault-tolerant architecture. The next chapter examines the current state of fault-tolerant architectures and how the logical building blocks can be translated to a fault-tolerant architecture. Additionally current qubit hardware platforms that are the most promising regarding fault-tolerant architecture implementation are examined.

Chapter 4

Fault-tolerant implementations and hardware options

As we have seen in the previous chapter, the logical resources are significant for implementing the modular exponentiation circuit for a $N = pq$ of 2048 bits. Recall that it is unrealistic to assume that running these large circuits will be without any errors, when these circuits are implemented on physical building blocks. To investigate the requirements for a quantum computer, which can run Shor's algorithm efficiently we need to discuss quantum-error-correction (QEC) techniques.

QEC techniques make the realization of large-scale practical quantum computers feasible by extending the possible execution time of a quantum state and by correcting errors that occur. Fault-tolerance architectures use these QEC-techniques. Fault-tolerance architectures are architectures that are designed to keep control over errors and as a result enable large-scale quantum computation. Fault-tolerance architectures are the topic of this chapter.

4.1 Fault-tolerant quantum computing

In a recent review by E. Cambell, B. Terhal and C. Vuillot different roads towards fault-tolerant universal quantum computation are discussed [18]. In this review the surface code is positioned as a promising architecture, because of its high-noise threshold, the requirements of only 2-dimensional (2D) qubit connectivity and a T-gate that is relative cheap to implement regarding the space-time overhead[1].

[1]The space overhead or spatial overhead is formulated as the number of physical qubits that are needed to form a logical qubit. The time overhead or temporal overhead is the difference in gate execution on a physical gate and on a logical gate [18].

A high-noise threshold is convenient, because the noise should always stay below the noise threshold. Fault tolerance can only be achieved if the physical-error rate[2] stays below the noise threshold [18, 19, 29]. If this is achieved, then arbitrary long quantum computations are possible with arbitrary accuracy[2]. This limits the impact of decoherence and faulty quantum gates [18] and make it possible to run Shor's algorithm successfully for a large N, such as 2048-bits. Note that the threshold is not a fixed value, but it depends on multiple variables: the type of error model(s) applied, the QEC method and architectural considerations, e.g. how the qubits can interact [18, 19].

4.1.1 Surface code

A fault-tolerant architecture can deal with errors as long as these errors can be identified. Once these errors are identified, errors can be corrected in any subsequent (measurement) operation. For the fault-tolerant architecture named surface code this is done using classical control software, e.g. Edmonds' minimum-weight perfect-matching algorithm. This algorithm works perfectly for sufficiently sparse errors, but fails when the density of errors increases [10]. Figure 4.1 gives an abstract representation of the steps taken by the surface-code architecture to perform error correction.

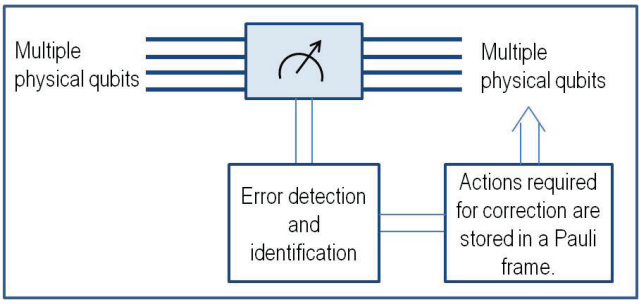


FIGURE 4.1: Abstract representation of the error-correction steps.

Error detection

In order to ensure that the measurements, which are needed to identify the errors, do not destroy the quantum system, the surface code applies a special type of procedure for measurements. The quantum systems are repeatedly measured using a complete set of commuting [3] stabilizers. An example for a two-qubit system is given in Appendix C. Stabilizers are operator products and measure multiple quantum systems simultaneously. Following this procedure the quantum system is forced into a simultaneous and unique eigenstate of all the stabilizers, which preserves (stabilize) the quantum properties of

[2]The physical-errors occur due to information leakage, measurement errors, control errors, decoherence etc.

[3]Commuting is a mathematical property, see Appendix C.

the state. The detecting property follows by comparing measurement results. If the measurement outcome is changed compared to earlier measurement results (eigenvalues), then one or more errors are detected. The measurement outcome is the eigenvalue and the measurement projects the quantum state into a stabilizer eigenstate [10].

The errors that are identified are stored in a Pauli frame. The Pauli frame functions as a memory that is updated each code cycle [30]. As with classical errors some quantum errors cancel each other out [10]. This gives the opportunity to optimize the error processing. It depends on the optimization choices that are possible on the hardware platform and the sequence of logic gate operations of the quantum circuit, when the information from the Pauli frame[4] is used to correct the errors [29]. Section 4.3 will provide information about the hardware platforms.

Logical operations

The surface code architecture is not complete if it is not able to facilitate logical-gate operations. The logical operations are built from the fault-tolerant Clifford+T-gate set. The Clifford+T-gate set is an important universal gate set [18].The Clifford gates are generated by the Hadamard gate, the S-gate[5] and the Controlled NOT (CNOT)-gate. However to achieve an universal gate set and to add quantum-computational advantage a non-Clifford gate should be added[6], this is the T-gate. The T-gate is a single qubit gate, represented by $T = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}$.

The properties of the surface-code architecture allow to create a logical qubit state and the logical Clifford-gate set. There are multiple options to realize these logical Clifford-gates. Some examples, derived from [10], are given below:

- A logical qubit is created e.g. by entangling multiple physical qubits, using physical CNOT operations and subsequent measurements.
- A logical qubit is initialized e.g. in an eigenstate of a qubit cut or hole. Holes are created by turning off measurement qubits.
- A logical Hadamard operation is created by physical Hadamard gates and SWAP[7] operations.
- The logical CNOT is created by e.g. a braid transformation[8], which can entangle two qubits.

[4] Note that after the information from the Pauli frame is used the Pauli frame is emptied to restart the collection of information.

[5] $S = diag(1, e^{i\pi/2})$

[6] Recall that with a universal gate set all quantum computations can be built.

[7] A swap is an exchange of quantum states between qubits [10].

[8] Braiding is a type of moving which enables entanglement. For surface coding a logical qubit hole is moved between two holes of a logical qubit [10].

To create a logical T-gate[9] multiple techniques are possible, however the most promising solution for this is magic-state distillation [18]. This magic-state distillation is also referred to as state distillation or ancilla state distillation [10].

Magic states are realized via a procedure that filters noisy quantum states to better quality states [18]. For this error-correcting codes and circuits, which involve single-control multi-target logical CNOTs, are used [10]. The T-gate is then realized using the magic states via a simple fault-tolerant circuit. Note that the T-gate is a few hundred times as costly as the Clifford gates in terms of space-time overhead and is therefore the most resource intensive building block [18].

Requirements for implementing a surface code architecture

The combination of state distillation and surface code is considered a competitive scheme and currently this combination is the most practical option to create a fault-tolerant architecture. If this practical most promising architecture is not suitable for requirements and other constrains, then alternatives are possible. However these alternatives are work in progress, e.g. work is needed to improve the noise threshold to be comparable with the noise threshold of surface code [18].

Requirements for physical implementations	**Values**
Minimum physical single-qubit gate fidelity	99%
Minimum physical two-qubit gate fidelity	99%
Minimum measurement fidelity	99%

TABLE 4.1: Requirements for implementing a surface-code architecture.

The requirements given to implement surface code on a physical platform are given in [10]. Table 4.1 gives a summary of these requirements.

Section 4.3 provides insight if the current specifications of hardware platforms are able to meet these requirements. First another requirement is derived, the number of physical qubits required to run Shor's algorithm on a surface code architecture.

4.2 A cost estimate for running Shor's algorithm

From the previous section we have learned that the combination of state distillation and surface code is considered a competitive scheme [18]. Appendix M of the paper on surface code by A. Fowler et.al. [10] gives an estimate of the amount of physical

[9]Note that depending on the surface-code implementation the S-gate also needs ancilla qubits for implementation, see [10].

qubits for running Shor's algorithm to factor a 2000-bit $N = pq$, using this competitive scheme[10]. This sections provides a summary and a reflection on the steps taken and results derived by Fowler et.al.

4.2.1 Circuit selection

Fowler et.al. starts with selecting a circuit that implements Shor's algorithm which is followed by a selection of the quantum adder circuit, see Figure 4.2. The selection criteria is to minimize the number of Toffoli gates, which in the end minimizes the required physical resources. More information about what a Toffoli gate is, can be found in Appendix C

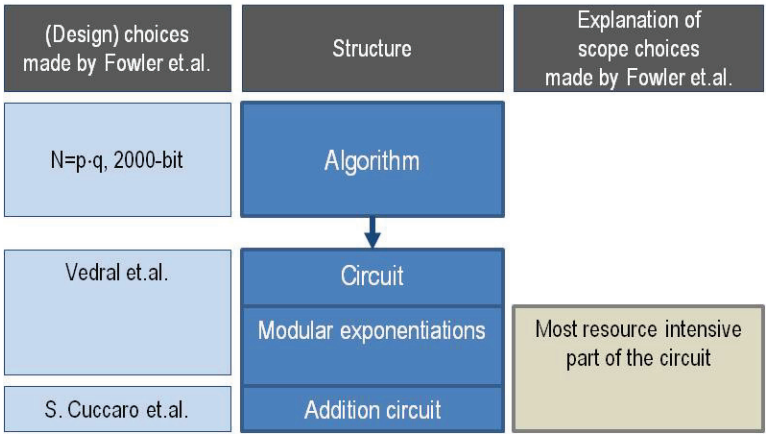


FIGURE 4.2: Fowler's choices to select the relevant quantum circuits.

4.2.2 Determine the required logic resources

The next step is to determine the number of required basic logic resources for the most resource-intensive part of the circuit. The logical resources are derived using the number of logic Toffoli gates the circuit requires for factoring a 2000-bit $N = pq$. Toffoli gates are build from T-gates, which need highly distilled ancilla states. The "factories" producing these ancilla states consume the most physical resources [10]. Figure 4.3 shows the steps Fowler et.al. take to estimate the number of logic ancilla states.

[10]Note: the magic states as refered to in [18] are called distilled logical ancilla states [10].

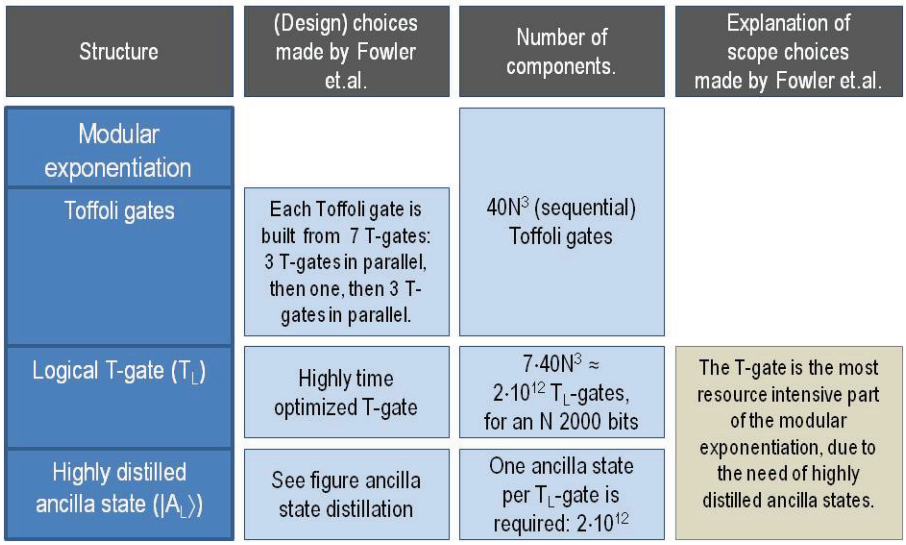


FIGURE 4.3: Scope and design choice made for estimating the required logical resources for the modular exponentiation circuit.

4.2.3 Determine the required physical resources

The ancilla states needed for the T-gate should meet two requirements to ensure that the modular exponentiation circuit makes negligible errors, which is needed for a proper functioning of Shor's algorithm. The first requirement is that the ancilla states are produced at a sufficient rate to keep pace to the modular exponentiation circuit. The second requirement is that the produced ancilla states should have a sufficiently small logical error rate. To ensure a good fidelity of the circuit the final logic error rate of $|A_L\rangle$ should be of the order $10^{-14} - 10^{-15}$.

This final logic error rate is achieved by multiple distillation stages. In the set up proposed in the paper by Fowler et.al. each distillation stage improves the error rate by $p_{new} = cp_{input}^3$, where c is a characteristic of the distillation circuit with value $c = 35$. Fowler et.al. assumes a $p_{input} = 0,005$ as the initial error rate as input for the distillation process.

The distillation circuit is implemented in surface-code to compensate for the errors, assuming an error probability per surface code cycle of $p = 0,01$, which results in a required fidelity of 99,9%. To minimize the total footprint[11] of the required circuit, the surface-code distance of the first stage is reduced. This is possible because the following stage will distill the remaining errors in order to achieve the required logic error rate.

[11] The footprint is defined as the number of logical qubits [10].

Each distillation stage has it own surface-code distance. The surface-code distance for each stage depends on six parameters, these are summarized in Figure 4.4. This figure also gives a summary of the steps to determine the required number of physical qubits to produce sufficient (high quality) ancilla's.

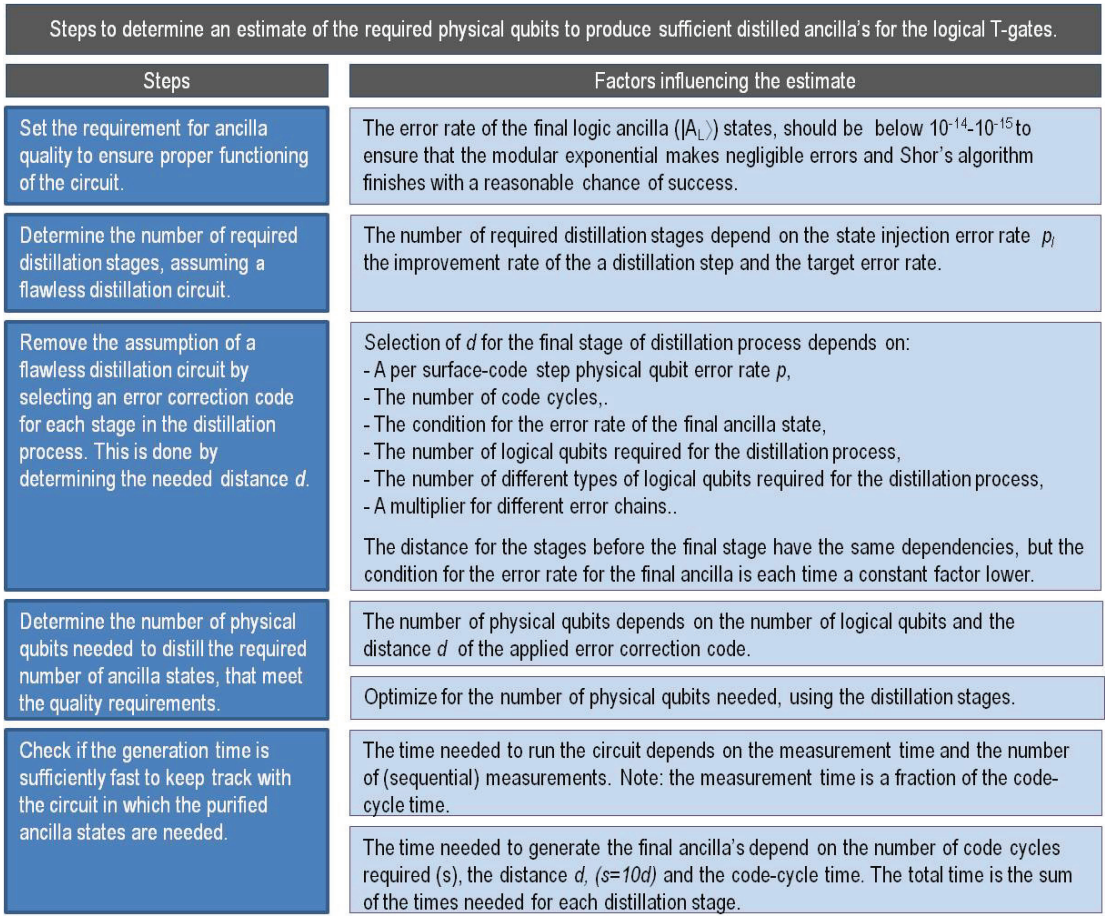


FIGURE 4.4: Steps to estimate the amount of physical qubits needed to produce the required ancilla states and parameters influencing this number.

The results derived in the paper by Fowler et.al. are summarized in the Figures below. Table 4.2 shows the number of logical ancilla states required for the T-gates that are the building blocks of the Toffoli gates. Recall that the Toffoli gates are the most resource intensive building blocks of the modular exponentiation circuit. The table shows that this number is significant higher as the required number of logical qubits for the remainder of Shor's algorithm.

Estimates for the required amount of logical components	**Values for N**	**Values for $N = pq$ for a size of 2000 bits**
Ancilla states	$280N^3$	$\approx 2,2 \cdot 10^{12}$
Qubits for the remainder of Shor's algorithm	$2N$	4000
Total number	$280N^3 + 2N$	$\approx 2,2 \cdot 10^{12}$

TABLE 4.2: Estimates for the required logical building blocks of Shor's algorithm.

The required number of physical qubits to produce these logical ancilla states is given in Table 4.3. The difference between the amount of physical qubits for the remainder of Shor's algorithm and the required amount for the production of ancilla states is less as seen in Table 4.2. This is due to the optimization options that can be applied in the ancilla factory, see Figure 4.4. The main optimization comes from the structure of the ancilla factor, each distillation stage requires less physical qubits, which gives the opportunity to reuse physical qubits. More information about this can be found in [10].

Estimates for the required amount of physical qubits	**For $N = pq$ of a size of 2000 bits**
For ancilla production	$\approx 2 \cdot 10^8$
For the remainder of Shor's algorithm	$\approx 1,4 \cdot 10^7$
Total amount	$\approx 2,1 \cdot 10^8$

TABLE 4.3: Estimates of the required amount of physical qubits for Shor's algorithm.

One of the main assumptions by Fowler et.al. was a T-gate that is highly time optimized. This means that the T-gate is completed in one measurement time t_M. The design of the Toffoli gate results in a completion time of $3t_M$. The modular exponentiation circuit has $40N^3$ sequential Toffoli gates this results in a completion time of $120N^3t_M$. Table 4.4 gives the assumed measurement time t_M and the time needed to factor a 2000bits $N = pq$.

Time	**Values**
Assumed code-cycle time	$200ns$
Assumed time required for a measurement (t_M)	$100ns$
Total time for the modulo exponentiation circuit	$120N^3t_M$
Estimation of the required time to factor a N of size 2000 bits	26, 7h

TABLE 4.4: A time estimation for Shor's algorithm using the Fowler et.al. set-up.

This section gave some indication for the physical requirements of running Shor's algorithm for a 2000-bit integer. This completed the list of requirements on the physical platform, this is summarized in Table 4.5. We also obtained insight in the assumptions made creating this indication. These assumptions are an injection error rate of $p_{input} = 0,005$ for the ancilla factories, a error rate per code cycle of $p = 0,01$, the measurement time is $t_M = 100ns$ and a surface-code cycle of $t = 200ns$ is possible.

Estimates for the physical implementation requirements	**Values**
Minimum physical single-qubit gate fidelity	$99,9\%$
Minimum physical two-qubit gate fidelity	$99,9\%$
Minimum measurement fidelity	$99,9\%$
Minimum physical qubit coherence times	$1 - 10\mu s^*$
Physical gate duration times	$10 - 100ns^*$
Estimate of the required amount of physical qubits to factor $N = pq$, with N of a size of 2000 bits.	$\approx 214 \cdot 10^6$

TABLE 4.5: Estimates of the requirements for implementing Shor's algorithm using a surface-code architecture on a physical platform.

The physical gate duration times and the physical qubit coherence times, both indicated with an asterix $(*)$ are derived using the assumption that the physical gate fidelities[12] are at least 99% and the classical processor operates at clock speed of 3 GHz, with the rounds of error detection applied at a speed of 106 to 107 Hz [10].

The next section will provide insight whether these assumptions can be realized on physical platforms. However, it was not possible to reproduce all values based on the information provided in the paper. For example: one of these assumptions could not be verified as its source was not published. The assumption, or design choice, that could not be verified was the highly time optimized T-gate see Figure 4.3. The paper provided another T-gate circuit [10], which uses one or three CNOT-gates[13]. These logical CNOT's are created using a technology called braiding. Braiding uses several move operations $(3 - 4)$, which each take $(1 + d)$ surface-code cycles. Each surface code takes 200ns, realizing a T-gate, as mentioned in the paper would take on average $(\frac{1+3}{2}) \cdot 4 \cdot (1 + d) \cdot 200ns$, where d is the applied code distance, d is an integer and $d > 0$. This takes significant more time to complete, than the assumed $100ns$, the time to complete a single measurement time operation. The design assumed for the T-gate impacts the estimated requirements significant.

[12] This includes the measurement fidelity, because gate operations and measurements are using a surface code architecture are strongly linked together [29].

[13] Note that the T-gate is a probabilistic gate. It produces 50% of the time a $T_L^\dagger$-gate, which can be converted to a T_L-gate using a S_L-gate and consist of two CNOT-gates and Hadamard gates, see page 31 and 32 in [10].

The error correction ability was tested by Fowler using simulation and error models. To examine the real practical feasibility of the surface-code architecture results from experiments needs to taken into account. However experiments applying large implementations of the surface-code architecture are not yet possible, because of the current availability of physical qubits. What the current number of available physical qubits per quantum chip is, is described in the next section.

4.3 Types of physical implementations

There are different options to create quantum mechanical states or qubits. Ion-trap qubits, superconducting qubits, nitrogen-vacancy-center qubits, photonic qubits and spin qubits are the most popular classes of technologies providing options to create a qubit [31]. All technologies have their advantages and disadvantages.

To select which technology is likely to implement a fault-tolerant architecture multiple parameters should be considered. The first parameter considered is the fidelity or precision of a two-qubit gate. The fidelity of a two-qubit gate should be above 99% to be able to implement a fault-tolerant quantum computer [31], see also Table 4.1. It is far from trivial that this requirement can be met.

However, in 2008 the ion-trap technology succeeded in meeting this requirement and, in 2014 the superconducting technology succeeded in meeting this requirement. From an evaluation of different qubit technologies made October 2016 follows that the other technologies are still in progress of meeting the requirement [31]. This narrows down the selection of technologies which are at this moment interesting to investigate for surface-code implementations.

4.3.1 Ion-trap qubits as the hardware platform for surface-code architecture

Ion-trap qubits are realized using ionized atoms. These ionized atoms are trapped in empty space by oscillating electric fields. The qubit is encoded in the direction of the spin of the atom's electrons with respect to an external magnetic field [31]. An example is visualized in 4.5.

Different spins or orbits correspond to different states. The quantum states are resilient to the environment [32]. This resilience also is shown by the qubit lifetime, the time that a qubit can utilize its quantum mechanical properties. For ion-trap qubits this is approximately 50 seconds.

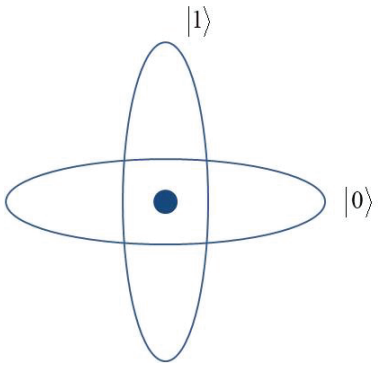


FIGURE 4.5: A visualization of ion-trap qubits with two different spin directions.

Gate operations used to be realized using laser beams. However if the number of qubits increases, then the number of laser beams would increase. As the ion-trap qubits have a size of approximately $1\mu m$ [31], this would mean that each laser beam needed to be controlled with an accuracy of at least $1\mu m$ [32]. A new approach for gate operations is to use microwaves. These microwaves are absorbed by the ion-trapped qubits only if there is a certain separation between the different states. Using the electric field to push the ions in the right magnetic field, changes the state [32]. A more detailed description of this technique can be found in [33].

The advantages of ion-trap qubits is that they can operate at room temperature and only need vacuum [31]. However, the time needed to perform a gate operation is larger compared to other qubit technologies. For laser controlled gate operations the time is approximately $50\mu s$ and for microwave controlled gate operations the time is approximately $3ms$.

Where the two-qubit gate fidelity of 99,9% is sufficient to use the ion-trap technology as hardware platform for the surface-code architecture, the number of qubits that are currently available does not meet the estimated requirements depicted in Table 4.5.

Another challenge should be mentioned. Ion-trap technology uses a 1D-qubit connectivity, instead of the 2D-qubit connectivity required to implement a surface-code architecture [29]. Therefore it is not possible to compare the estimated requirements for running Shor's algorithm one-on-one to the specification achieved by researches developing ion-trap technologies.

4.3.2 Superconducting qubits as the hardware platform for surface-code architecture

There are multiple types of superconducting qubits. An overview is provided by Wendin [34]. Superconducting qubits are realized in electronic circuits made of superconducting materials. These circuits should be kept in cryogenic temperature ($\sim 10mK$) to reduce the electric noise, which influences the quantum state. The gate operations on qubits are realized using microwaves and electric drivers.

Many type of superconducting qubits, like transmons, flux qubits and phase qubits are built using Josephson junction based qubit circuits. In Figure 4.6 a basic equivalent circuit for Josephson junctions based superconducting non-linear oscillator qubits is depicted. Josephson tunnel element is depicted as a crossed box with an inductance L_J and a capacitance C_J. Parallel to the tunnel element a capacitance C, and an inductance L [35].

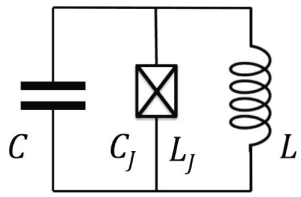


FIGURE 4.6: A basic equivalent circuit for Josephson junctions based superconducting qubits, derived from [34, 35].

An advantage of superconducting qubits is that the two-qubit gate operation time, of approximately $50ns$, is relatively fast when comparing the other qubit technologies. However superconducting qubits need to be operated at very low temperatures to maintain the superconductivity and to minimize noise. To realize this extreme cooling expensive dilution refrigerators need to be used [31].

The two-qubit gate fidelity of 99,4% is sufficient to use the superconducting technology as hardware platform for the surface-code architecture. The paper of Campbell reports several efforts with superconducting qubits to build a logical qubit using the surface-code architecture [18]. However the estimate of the required physical resources in Table 4.5 assumes a fidelity of 99,9%. A lower fidelity, relates to a higher error probability per code cycle (p), which is one of the parameters influencing the estimate of the required amount of physical qubits. A higher error probability p will result in an increase of the code distances d in order to achieve the requirements for successfully running the modular exponentiation circuit. A higher code distance d will increase the number of physical qubits per logical qubit, and therefore increase the total number of physical qubits required [10].

4.3.3 The current number of physical qubits on a quantum chip

The current number of physical qubits on a quantum chip differs per research lab. Increasing the number of available qubits seems like a race. One of the current objectives is to achieve the level of 50 physical qubits level, e.g. IBM is currently testing a 50 qubit prototype based on superconducting qubits [36]. Also Intel released a 49 superconducting qubit chip [37], and Google announced their hope to release a 49 superconducting qubit chip [38]. This amount of qubits is supposed to be required to achieve quantum supremacy.

Quantum supremacy is referred to as the situation that the output of a quantum computer cannot longer be simulated on a classical computer. Classical computers can be used to simulate quantum computers. The largest super computer, with $\sim$ 10 Petabytes of memory is supposed to simulate $\sim$ 48 qubits [31]. Having 50 qubits is supposed to enable quantum supremacy. However, when the level of 50 qubits is achieved the number is still insufficient to be able to run Shor's algorithm to factor a $N = pq$ of 2048 bits.

4.4 Closing the gap

There are efforts to close this gap as a modular architecture is developed to form a large-scale quantum computer out of many small modules [31]. The size of a large-scale quantum computer, using this modular architecture for ion-trap technology, is estimated to approach the size of a football stadium [32].

Currently an ion-trap quantum computer demonstrator is being built at the university of Sussex, which will take 2 years to build starting from 2016. To have a large-scale quantum computer available prof. Hensinger gives an estimate of 10+ years starting in 2016 [32]. Professor Hensinger is associated with the university of Sussex, this university participates in the UK National Quantum Technologies Program.

Currently IBM, with the announcement of a prototype of 50 qubits, is the front-runner for superconduction qubits. Note that D-Wave also uses superconducting qubits and has a currently a larger amount of qubits, but the D-Wave quantum computers do not facilitate circuit based quantum computation, see Appendix D, and are therefore not included in this evaluation.

However the identified gap can also be closed by progress in error-correction schemes and T-gate generation, which will reduce the fault-tolerant costs. Reducing the demand side and increasing the supply side, see Figure 4.7, both contribute to closing the gap.

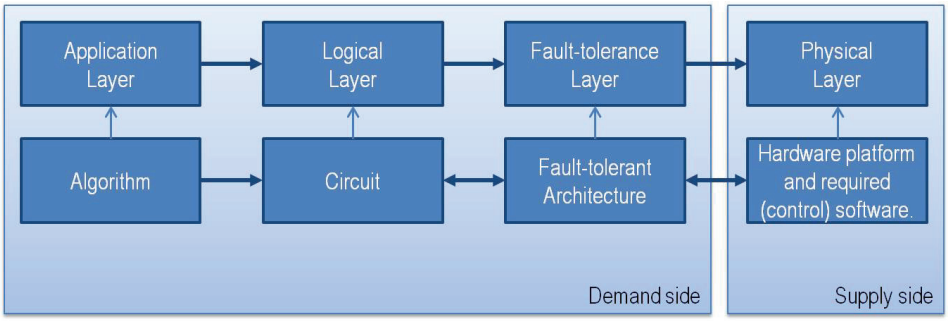


FIGURE 4.7: Layers grouped in a demand side and a supply side in relation to the physical resources. Both sides contribute to closing the gap.

Observe that the focus of this thesis is on breaking RSA by applying Shor's algorithm, but other algorithms can be broken by applying Shor's algorithm, see Chapter 2. This will result in other gaps as the demand side will results in different requirements.

4.5 Summary

In this chapter we have seen that a fault-tolerant architecture is necessary to implement Shor's algorithm in order to factor a N of size 2048 bits and to handle errors, which occur during computation.

Currently the most promising fault-tolerant implementation is the surface-code architecture with magic state distillation to produce the necessary T-gates. However fault-tolerant implementations come at a price. From the paper by Fowler on the surface-code a estimation of the amount of physical resources required for running Shor to factor a 2000 bits $N = pq$ was investigated.

This estimation was used to determine the most likely hardware implementations. Comparing the required physical resources with the available resources led to a significant gap between the demand and the supply side.

There are many efforts to close this gap. The next milestone for the supply side is to create a quantum chip with 50 physical qubits in order to show quantum supremacy. However scaling up the number of controllable physical qubits or increasing the fidelity of the physical gates is not the only pad to close the gap. Progress in error-correction schemes or more efficient ways to create T-gates will also contribute in closing the gap.

Chapter 5

Reflection on the quantum threat

This chapter provides a reflection on the risk event defined for this thesis: The event that Shor's algorithm can be run on an available quantum computer and by running the algorithm it is able to break RSA with a 2048 bit key (RSA-2048). The goal of this reflection is to provide input for a risk assessment, to enable the risk assessors to make informed decisions on how to act regarding the quantum threat.

First we provide a reflection on the likelihood and how this might evolve over time. Second a method is presented to translate the technical impact as formulated by NIST, see Chapter 2, to how this relates to the strategic risks of an organization. The reflection on the impact is closed with a brief summary on how the quantum threat can impact a highly ICT-dependant society.

In the section about how to act regarding the quantum threat, four options are described and reflected upon. Additionally a brief introduction of two types of quantum-safe solutions is provided as possible options for risk controls. The chapter is closed with the introduction of a monitoring framework to monitor the quantum threat.

5.1 Likelihood

Chapters 3 and 4 provided information about the requirements to run Shor's algorithm and break RSA-2048 and information about the current state of quantum computers. The focus of Chapters 3 and 4 was on the approach that is currently believed the most likely approach, see Figure 5.1. In regard of this most likely realization, the gap analysis showed that running Shor's algorithm for a significant key size demands significantly more than the current supply side can offer. An easy conclusion would be that the likelihood of the quantum threat is negligibly small.

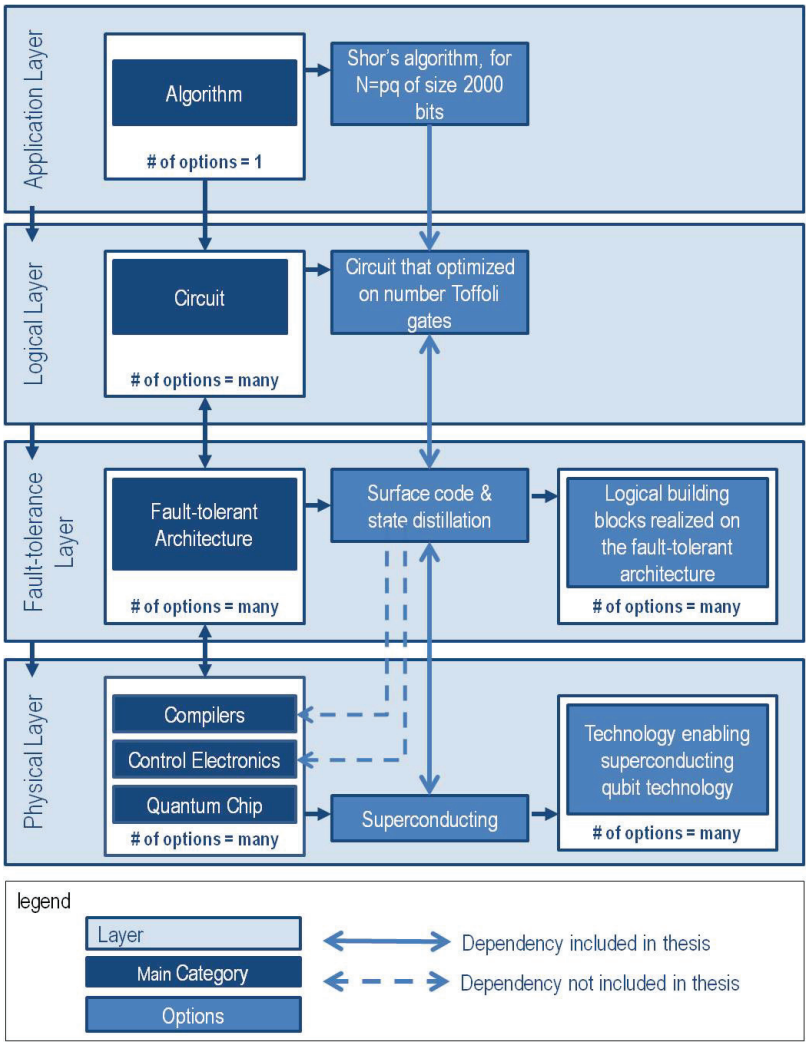


FIGURE 5.1: Visualization of the options and choices made.

During the information gathering about the most likely path to realize a quantum computer that can run Shor's algorithm, multiple different options for realization were found on each layer[1]. Some options were already mature enough to use in an experiment while others needed theoretical improvement to gain sufficient advantage over the current most likely alternative, like described for fault-tolerant architectures in Chapter 4.

This significant amount of variables makes it difficult to predict the exact year that a large scale fault-tolerant quantum computer becomes available. This is also shown in the estimates given by experts. The most precisely formulated estimate is probabilistic and is 1/7 chance on availability in 2026, 1/2 chance on availability in 2031 [23]. It is however not possible to evaluate the exact calculation of this estimate, because this is proprietary information and not available for review.

[1]Note: the application layer has only one option, because we selected Shor's algorithm as research scope.

For the likelihood we can conclude that the amount of variables makes it unrealistic to formulate an exact, non-probabilistic, estimate when a large-scale fault-tolerant quantum computer becomes available. However threats dynamically evolve over time, this is also true for the quantum threat and the possibility to estimate an arrival time of a large-scale fault-tolerant quantum computer. Three observations should be taken into account:

1. What is currently believed the most likely path might change to as less likely path. At this moment in time other paths are possible, see Figure 5.1. These paths will result in other demand-supply gaps, which in time might be more realistic to close. This should be taken into account when evaluating the likelihood.
2. The different quantum research fields, which contribute to closing the gap, are not static but dynamic. These fields are continuously evolving and solving problems supported by significant funds, see [3, 39]. This should be taken into account when evaluating the likelihood.
3. This thesis focused on Shor's algorithm breaking RSA-2048. However Shor's algorithm also effects other cryptographic algorithms see Table 2.1. This will affect the required physical resources and will result in a different gap. The different gaps for different cryptographic algorithms should be taken into account when evaluating the likelihood of the complete quantum threat.

Uncertainty about how the likelihood evolves over time is not a problem if the impact of a risk is low. However the impact, that occurs when the quantum threat materializes in a risk, can be severe depending on the context of an organization. What should organizations do regarding this uncertain evolution of the likelihood if the impact cannot be ignored? This issue is addressed in Section 5.3. First a brief reflection on the impact is provided.

5.2 Impact

As discussed in Section 2.2 and Table 2.1 the impact of quantum computing in general is significant. The impact of breaking and crippling crypto-systems is very clear for technically oriented people, as crypto-systems are a crucial part of almost all IT-systems[2]. However, as cryptography is hidden within IT-infrastructures, the debate about the quantum threat benefits from translating this technical impact of quantum computing to the risks managed at board-room level.

[2]ETSI provides a list with parts of general IT infrastructure and applied crypto-systems [6].

5.2.1 Method for translating the technical impact to strategic risks

To translate the technically formulated impact to how it effects the strategic risks of an organization the following reasoning should be applied.

Organizations transform their unaccepted risks into accepted risks by applying risk controls. A single unaccepted risk can be mitigated to an accepted risk by one or multiple risk controls. Crypto-systems are one of the options to mitigate a risk. In practice this is a frequently applied risk control.

If the applied crypto-systems are broken or crippled, then it is not longer an adequate risk control. Depending on the other applied controls the risk level is changed. If the risk level is reduced to an unaccepted risk level, then the impact is not longer only formulated as breaking or crippling crypto-systems, but also formulated as not longer being able to control an organization's risk. In the box below an example is given.

Example - translating to GDPR compliance.

A concrete risk for an organization is non-compliance to regulation. For example non-compliance to the EU's General Data Protection Regulation (GDPR), which is part of the data protection reform of the European Commission [40]. The GDPR will enter into force on 24 May 2016, however it shall apply from 25 May 2018. Under EU law, personal data can only be gathered for legitimate purposes and under strict conditions. Organizations which collect and manage personal data must respect certain rights of the data owners and must protect it from misuse [40].

Protecting the processing of personal data includes protecting it when it is transferred and when it is stored. An imported form of protection is provided by crypto-systems. When these crypto-systems are not longer trustworthy, the protection of the personal data becomes inadequate, which results in not being able to comply to EU regulation[a].

This example translates the impact of the quantum threat from breaking or crippling crypto-systems to non-compliance to regulation.

[a] In this particular case also a financial penalty can given.

5.2.1.1 Actors involved in strategic risk-management for listed companies

The risk of not being compliant to relevant regulation is a risk that is addressable at an organization's management board, as being compliant to the relevant regulation is the responsibility of the management board. The Dutch Corporate Governance Code (DCGC) of 2016 provides guidance[3] for effective cooperation and management of listed companies. Not only the management board is defined as an actor regarding the strategic

[3] Starting January 1, 2018 the Dutch law requires all Dutch listed companies to report on the compliance with the DCGC [41].

risk-management, but also the supervisory board. Their responsibility is to supervise the effectiveness of the organization's internal risk-management and control systems. For these tasks reports, from the audit committee are used [41].

The audit committee monitors the management board on several items. One of the items is the application of ICT by the company, including risks related to cyber security [41]. If a translation can be made from the technical impact of quantum computing[4] to not adequately controlling a strategic risk, then it is justified for the audit committee to include the quantum threat in the reports provided to the supervisory board. Recommendation on how organizations act regarding the quantum threat is the topic of Section 5.3.

5.2.2 Impact of the quantum risk to society

The scope of the previous section is limited to organizations, however because our society increasingly depends on the benefits from ICT [42], the quantum threat also should be evaluated for our society. When the quantum threat materializes in a risk it has positive consequences and negative consequences.

The quantum computer is supposed to be of great value for research in the field of medicine, material science and energy [3]. But also promises to solve computational problems in the research areas of transport, logistics and artificial intelligence [4]. These promised benefits result in large investments worldwide in the different fields related to quantum computing [3, 39]. To enjoy the benefits from the quantum computer as a society, the negative consequences should be minimized when possible.

ETSI published in [6] a list of IT-infrastructure-building blocks that are vulnerable, when there is a large-scale fault-tolerant quantum computer available. This list includes the certificates used for Public Key Infrastructure (PKI) issued by commercial CAs, Digital Signatures used for Secure Software Distribution, Secure Email (i.e. S/MIME), Virtual Private Networks (i.e. IPsec) and, Secure Web Browsing (SSL/TLS). The used security protocols SSL/TLS, SSH and IKE/IPsec rely almost exclusively on key exchange using RSA, Diffie-Hellman, or Elliptic Curve Cryptography. The certificates used for PKI and the certificates used for S/MIME contain RSA public keys [6].

All these IT-infrastructure-building blocks are used to enable society to benefit from ICT. To name some of the benefits: online shopping and banking, online access to your health-care test results, online registration for social funds for citizens, online access your kids daycare reports or student files, providing citizens online trusted information

[4] Recall this refers to the NIST table and can be summarized as breaking or crippling crypto-systems

about taxes, but also about calamities, etc. Not being able to trust these building blocks supporting these and other online services cripples society. In order to prevent this negative consequence for society, society should act.

5.2.2.1 Actors involved

The question remains who should act and what should be done. To answer the who-question the second cyber security strategy (NCSS 2) of the Netherlands provides an answer. Three type of actors are defined in the NCSS 2, citizens, businesses and government. The responsibilities of these actors follow the underlying fundamental principle that the responsibilities that apply in the physical domain should also be taken in the digital domain [42].

Citizens are responsible for some skills using ICT like browsing on the internet and apply some basic cyber-hygiene. This cyber-hygiene includes installing updates, but not knowing how these updates work or what type of encryption is involved [42]. Using the right type of encryption is a responsibility of the business or the government who provides the ICT-service or software used by the citizen.

Businesses, defined as providers of ICT networks and services or other ICT-based services, have a specific responsibility with respect to their clients. This responsibility is also referred to as the duty to care. The NCSS 2 states that this responsibility is preferably achieved by means of self-regulation [42]. The approaches described in Sections 5.2.1 and 5.3 give businesses the tools to act on the quantum threat.

The government is responsible for its own ICT services and as a regulator and facilitator responsible for providing adequate information. Steps to provide information are already taken, e.g. the report on post-quantum cryptography from the National Cyber Security Center (NCSC) [7]. However the role of the NCSC can be expanded as will be described in Section 5.4.

One of the core tasks of the government is the prevention of social disruption [42]. Considering the negative impact on society, the government should at least monitor the progress of risk mitigation for the vital sector. If this progress is insufficient, then the government should act in a controlling manner. Note that the progress depends on the progress the availability of standardized quantum-safe solutions, see Section 5.3.6. A framework for monitoring the quantum threat is proposed in Section 5.4.

5.2.2.2 Management of the interdependence between the actors involved

To manage the increasing dependency between the three actors and the pursue of a balance between security, freedom and social-economic benefits, three management areas are defined in the NCSS 2: (self)regulation, transparency and knowledge development. For the quantum threat all three areas are relevant. As self-regulation refers to the development of standards, this also applies to standards for quantum-safe solutions, see Section 5.3.6. Transparency is about sharing applied solutions, this could for example be applied for how businesses and the government monitor and act on the quantum threat.

Knowledge development and sharing are the most important management areas regarding the quantum threat, because of the high impact on society, the uncertainty on when the likelihood increases significantly and given that the mitigating measures are in the process of standardization, see Section 5.3.6. How to act on the given uncertainties and creating understanding between the different positions regarding the quantum threat helps to determine the most effective risk-treatment plan for society.

However the quantum threat does not only have consequences for the Netherlands but for all countries that heavily depend on ICT. The Netherlands could use its ambition, as formulated in the NCSS 2, to play a prominent role in the search for new coalitions in which all parties involved are represented in order to reach internationally accepted standards related to actions in the digital domain to mitigate the negative consequences of the quantum threat. This should be part of an effective risk-treatment plan, because of the internationally interconnected character of our digital infrastructure.

5.3 How to deal with the quantum threat?

As discussed in Section 5.1 the current likelihood of the risk event is small and it is not possible to determine an exact date when the threat materializes. This uncertainty on how the likelihood evolves does not justify a long resource-intensive process of developing a plan of action or risk-treatment plan at this moment in time. However considering the impact, the quantum threat can not be ignored. In this section four options are discussed. The first relates to actions when the screening of risks results in insignificant risks for an organization. The second and third option relate to a more detailed risk assessment. The second is to determine the risks and the third includes risk treatment actions. The last option is to treat the quantum threat without further risk assessment.

5.3.1 Option 1: Delaying the action plan

When after screening the risks it is decided that the quantum threat only poses an insignificant risk regarding the organization's responsibilities, then limited action is required. However when screening the risks, two important factors should be taken into account:

- Data collection and storage possibilities.
  Current technologies enable large-scale copying of information and storing this information for a long time. This is also possible for encrypted information. If this encrypted information is stored long enough, then eventually a large-scale quantum computer is able to decrypt it [11]. Some of this decrypted information may have lost its value over time, but there is also information that retains its value for longer periods of time, like health-care records and state secrets.

  This scenario is important for all organizations that work with valuable information that has a long retention time and which damages the organization when the information is leaked and decrypted even after a longer period of time.

- Migration time.
  NIST states that it has taken 20 years to deploy our current public-key infrastructure, and it will take a significant effort to ensure a safe and non-disruptive migration [5]. The migration time differs for each type of organization. The choice for risk-assessment and mitigation approach will determine if the necessary efforts can be balanced, see Section 5.3.3.

  The migration time for the organization should be taken into account when screening the risks regarding the quantum threat.

If after considering these factors delaying is still the best option, then limited action is required. The action that is required, is regular monitoring and reviewing the risk.

5.3.2 Option 2: Mosca & Mulholland's Methodology for quantum risk-management

For organizations that like to perform a risk-assessment to determine their risks related to the quantum threat two options are provided. The first is described in this section, the second in Section 5.3.3.

In January 2017 the Global Risk Institute published a Methodology for Quantum-Risk Assessment (QRA) consistent with risk-assessment models such as NIST, and using Mosca's "x, y, z"- quantum - risk model [22] accompanied by several recommendations.

The main recommendation is to act now by including Mosca & Mulholland's QRA to the regular risk management process. The QRA that is proposed by Mosca & Mulholland has six phases and is summarized below:

- **Phase 1** Identify and document sensitive and valuable information assets, and their current cryptographic protection including the type of encryption, the key generation method, key storage and how they are applied. Also include the origin of the related tools and appliances.
- **Phase 2** Research the state of emerging quantum computers and quantum-safe security measures. Estimate the timelines for availability of these technologies. Influence the development and validation of quantum-safe security measures. This is a continuous process.
- **Phase 3** Identify threat actors and estimate their time to access quantum technology. This is element "z" in Mosca's model.
- **Phase 4** Identify the lifetime of your assets: how long should the information kept secure. This is element "x" in Mosca's model. And identify the time required to transform to a quantum-safe solution. This time is element "y" in Mosca's model.
- **Phase 5** Determine the quantum risk by calculating whether business assets become vulnerable before the organization can move to protect them. If $x + y \leq z$, then you can determine when it is time to start preparing. If $x + y > z$, then you are already at risk.
- **Phase 6** Build a road-map to a quantum-safe company. Include a prioritized list of activities that are required to maintain awareness and include activities for migration to a quantum-safe state.

For Phase 2 it is recommended to have a dedicated team of quantum experts or to have a relationship with an organization specializing in quantum technology [22]. At Phase 3 it is also recommended to investigate if new threat actors might emerge once quantum computing becomes a reality. It is also recommended to understand which of your vendor's IT products will be affected and what are their preparations to manage this risk. The last recommendation is to evaluate the state of the quantum-migration planning of your network-and-security vendors as part of your current procurement processes.

5.3.2.1 Reflection on Mosca & Mulholland's quantum-risk assessment

Mosca & Mulholland's QRA is an (information) asset-based approach, which is common for many risk assessment methodologies [22] and will fit easily in many used risk-assessment processes. This approach will result in a thorough analysis of which assets are at risk.

The approach needs a current inventory of sensitive and important assets, as described in Phase 1 of the QRA. A current inventory of sensitive assets is also useful for other security-related processes. However, if such an inventory is not already in place or when the required information on their current cryptographic protection is not in place, then adding this information and keeping it up-to-date will require significant investments. Considering that the current estimates of the risk materializing are not in the distant future [5], these investments can not be justified for all organizations.

Another possible drawback of this method is that if the results from Phase 5 determine that you are already at risk, then this approach will slow down the response time as it depends on information obtained from the previous phases. In the next section a more pragmatic approach is proposed, this approach will reduce the amount of efforts needed for Phases 2 and 3.

5.3.3 Option 3: A pragmatic approach

This section proposes a pragmatic approach for assessing and acting on the risks regarding the quantum threat. The main objective of the pragmatic approach is to find a balance between the required investments for assessing and acting, and the current uncertainty about when the risk materializes.

Check if the technical impact translates to strategic risks

This pragmatic approach does not identify the risks by evaluating the sensitive and important information assets, but identifies if the strategic risks of an organization are impacted by the quantum threat. The impact on the strategic risks is evaluated using the method provided in Section 5.2. The strategic risks of an organization are already formulated as required by regulation, e.g. in the DCGC [41].

[5] Recall from Section 5.1, that the most exact estimation is: 1/7 chance on availability in 2026, 1/2 chance on availability in 2031.

The required actions depend on the impact on the strategic risks

After applying the method for translating the technical impact to the impact on strategic risks of Section 5.2.1, roughly three scenario's can occur.

- Controlling[6] the strategic risks requires applying crypto-systems,
- Controlling the strategic risks does not require applying crypto-systems, but the organization depends for its business operations on IT provided by third parties,
- Controlling the strategic risks does not require applying crypto-systems, and the organization does not depend on IT provided by third parties.

If controlling the strategic risks does require applying crypto-systems, then this strategic risk should be assessed again using the scenario that a large-scale quantum computer is available. If the remaining controls are not sufficient to reduce the risk to an acceptable level, then quantum-safe solutions as described in Section 5.3.6 should be applied. The urgency for migration depends on the context of the risk. Monitoring the risk helps to determine if the urgency changes. Input for the monitoring of the quantum threat can be found in Section 5.4.

In this first scenario it is possible that part of the IT is outsourced or managed by a third party. In that case the current contracts should be evaluated to ensure that these contracts support risk mitigation to quantum-safe solutions.

If controlling the strategic risks does not require applying crypto-systems, but the organization depends for its business operations on IT provided by third parties, then the organization should consult their relevant suppliers to understand how these third parties ensure compliance to standards and regulations (like e.g. GDPR) regarding the quantum threat. Together should be decided if strategic risks are impacted and if a mitigation plan should be formulated, to ensure that the identified strategic risks stay on an accepted risk level.

If controlling the strategic risks does not require applying crypto-systems and the organization does not depends on IT provided by third parties, then no direct urgency is identified and only the steps as described in the next section are required.

Create a quantum-safe baseline for the future

All three scenario's described above need to ensure that the quantum threat does not introduce new unknown risk for the organization. To ensure this the organization should make the following strategic decisions:

[6] controlling refers to mitigating to an acceptable risk level.

- The procurement department should add in every new IT contract a paragraph that states that standards for crypto-systems must be followed including the recommendations made on quantum-safe solutions by the standardization institutes.
- New business strategies should be assessed using the scenario that a large-scale quantum computer is available.

Making the first decision ensures a smooth and secure migration to a quantum-safe organization, including all IT assets provided by suppliers. The second decision ensures that if new business strategies are rolled-out with a quantum-ready design, making sure that this does not result in additional investments after implementing the new strategy.

5.3.3.1 Reflection

This approach differs from Mosca & Mulholland's QRA method as it is not an (information) asset-based approach. The pragmatic approach starts with determining the impact on the organization's strategic risks. If strategic risks are effected, then a mitigation plan should be made. IT-infrastructure components that are not related to a strategic risk are mitigated with a low priority via standard IT procedures initiated by contracts with suppliers as part of creating a quantum-safe baseline for the future.

This approach saves scarce resources and unjustifiably large investments. This creates balance between the current uncertainty about when the risk materializes and the investments required to investigate the risk and act on it.

5.3.4 Option 4: Risk-treatment without assessment

Acting on the quantum threat without any risk assessment is not advised. Part of the risk-analysis phase is to assess the controls. This assessment include the current controls, but also new controls which are supposed to mitigate the risk level to an accepted level. Two important factors for the assessment of controls are: if the controls operate in the manner intended and if they can be demonstrated to be effective when required. Standardization of controls plays an important part in this matter. However the current standardization process is still ongoing for post-quantum cryptography, see Section 5.3.6.

Starting mitigation without a finished standardization process needs additional risk assessment on the possibility of introducing new risks. If the data involved is not valuable for a long period of time, then it is recommended to monitor the availability of standardized quantum-safe solutions.

5.3.5 ICT organizations

ICT businesses as formulated in the NCSS 2 or organization's that have in-house development of ICT-systems and services have to take additional actions to the actions described for the first scenario in Section 5.3.3. These actions include training employees to understand the quantum-threat and informing the employees about the organizations strategy regarding the quantum-threat. This enables employees to answer questions from customers and start embedding quantum-safe solutions in the road-maps they maintain.

5.3.6 Mitigating measures - Quantum-safe solutions

Fortunately not only the scientific fields related to using and building a quantum computer are making big steps, but also the scientific fields protecting IT against the threats of the quantum computer. In the quantum field, a quantum attack is an attack for which an adversary uses the computational powers of the quantum computer. There are two main solutions protecting against quantum attacks, the field of post-quantum cryptography and the field of quantum cryptography.

Post-quantum cryptography

The field of post-quantum cryptography studies and develops classical algorithms. These algorithms are not based on mathematical problems like factoring and discrete logarithms, but on mathematical problems that are believed to be secure against quantum attacks [24]. Because post-quantum cryptography are classical algorithms, they are supposed to be easily migrated in current systems [7].

On November 30, 2017 NIST's Computer Security Resource Center (CSRC) closed the submission for public-key post-quantum cryptographic algorithms. Currently time lines for selecting one of more algorithms and realizing the first draft standard are estimated to be finished in the year 2022-2024 [43].

Depending on the urgency of the identified impacted strategic risk, this timeline could be a problem. For example when the information that needs to be protected has a long retention time regarding the information value and the scenario that an adversary collects and and stores this information is considered likely. If this is the case, then now is the time to address this using the knowledge development and sharing management areas as defined in the NCSS 2. By addressing this problem other adequate mitigating measures can be defined and supported by all parties.

Quantum cryptography

Contrary to post-quantum cryptography the field of quantum cryptography uses quantum-mechanical properties to obtain security. The main solution in the field of quantum cryptography are quantum key distribution (QKD) systems. Quantum key distribution systems use properties of quantum mechanics to continuously establish a secure symmetric key between two parties [2]. This quantum-safe key can be applied in symmetric crypto-systems. By applying a large key size as prescribed by NIST [5] and applying a high refresh rate of the key a quantum-safe solution can be created.

QKD systems are already commercially available for more than a decade [44]. These QKD systems need a optical fiber to transmit photons between two parties for the key establishment[7]. The distance between the two parties is limited to approximately 100 kilometer and the achievable key rate depends on the quality of the optical fiber and the distance between the two parties [44]. However in September 2017 an article was published in Nature on a satellite-to-ground QKD link achieving a kilohertz key rate at a distance of 1200 kilometer in free space. This result is a step towards global scale quantum networks [45].

5.4 A framework for monitoring the quantum threat

Monitoring helps to determine the evolution of the estimated time of arrival of a large-scale quantum computer or the variable z from Mosca & Mulholland's QRA method. The information obtained from monitoring the quantum-threat helps an organization to adjust their mitigation time lines when necessary and to keep in control of the identified risks.

The monitoring framework can be derived from the findings in Chapters 3 and 4. Because the focus of this thesis was on Shor's algorithm for breaking RSA-2048, some additions are made to make it possible to monitor the complete quantum-threat including the quantum-safe solutions. Additions are made in the application layer, the physical layer and the group quantum-safe solutions is added.

In the application layer the focus on Shor's algorithm is changed to a more general formulation: quantum algorithms that pose a threat to cyber security. Currently efforts on realizing new quantum algorithms are made [36]. These algorithms could intentionally or unintentionally result in positive or negative consequences for cyber security. Monitoring this field helps to detect new threats. In the physical layer the user-friendliness

[7] Note the optical fiber is needed as a quantum channel. No active components are allowed as these destroy the quantum states encoded in the photon's. Also a classical channel is needed for authentication purposes.

of the quantum computer's interface is added, as will be explained below. Figure 5.2 shows the resulting monitoring framework for the quantum-threat including the most important topics to monitor.

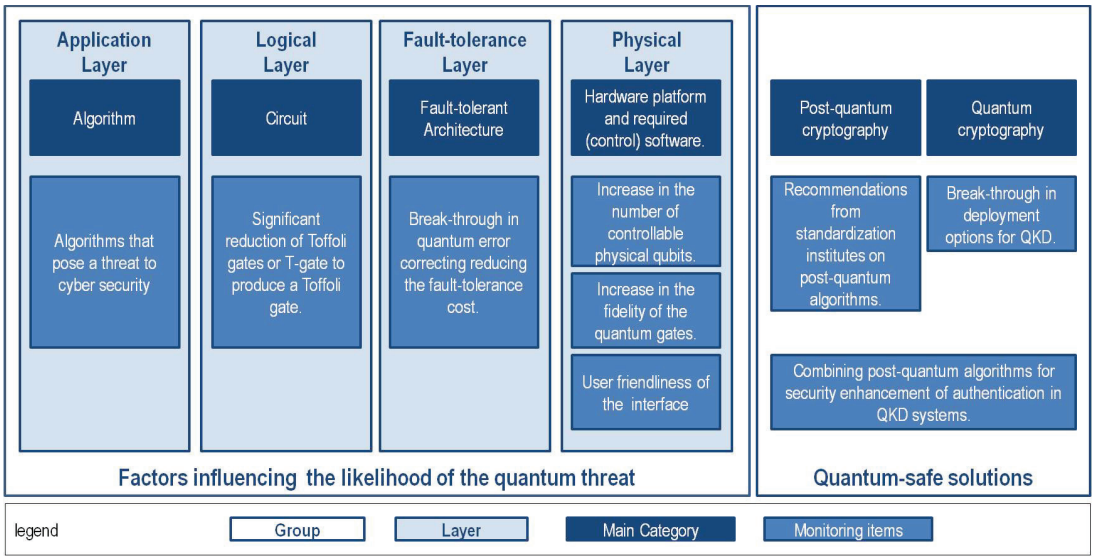


FIGURE 5.2: Visualization of the framework for monitoring the quantum threat.

User-friendliness of the quantum computer as monitoring topic

User-friendliness of the interface is very important for the further development of the positive consequences of the quantum computer. User-friendliness of the interface of a quantum computer is also included as monitoring topic for the quantum-threat as it impacts the type of threat actors that might use the quantum computer to break or weaken applied crypto-systems or other relevant part to cyber security. Adding this topic provides organizations information to estimate the required knowledge that a threat actor needs in order to use a quantum computer as part of their attack scheme.

Actors involved in monitoring the quantum-threat

Monitoring the quantum-threat is an important part of being able to adequate anticipate the negative consequences of the arrival of a large-scale quantum computer. Considering the current uncertainty about the time of arrival, all organizations in a highly ICT dependant society need the information obtained from monitoring.

Monitoring the quantum threat will not be a straightforward task, because of the multiple relevant variables. Support from an informed party is needed [22]. The NCSS 2 describes the role of the National Cyber Security Center (NCSC) as expert authority, providing advice to the involved private and public parties, both when asked and at its own initiative [42]. This includes to advise on detected major vulnerabilities or at crisis

situations. Monitoring the quantum-threat does not fit this description perfectly, but it does not deviate much from the role of the NCSC.

The NCSC is therefore the best candidate to provide private and public parties with information obtained from monitoring the quantum-threat. This information will also support the NCSC in planning which type of advice is needed at what moment in time. It will also provide the NCSC tools for deciding when it is time to escalate, e.g. when the progress in the vital sectors is insufficient to prevent disruptions in society as described in Section 5.2. The NCSC could use the quantum knowledge-base from universities and research institutes both national and international to get up-to-date information about the topics as defined in the monitoring framework.

5.5 Summary

In this chapter we reflected on the likelihood of the quantum-threat to materialize in a risk. The significant gap between the demand and supply side as formulated in Chapters 3 and 4 resulted in concluding that it is unlikely that this gap is closed in a short period of time. However this does not conclude that currently no risk-management regarding the quantum threat is required, it only concludes that it is likely that we have time to prepare. Preparation is necessary considering the following cases.

First, RSA is not the only vulnerable algorithm. Other algorithms will require another set of physical requirements, which will result in different gaps. Second, post-quantum algorithms, which are considered a likely risk control, are still in the process of standardization. Third, a case that should be considered is that encrypted information can be collected and stored for a long time. When there is a quantum computer, this stored information can be decrypted. If the information is still considered valuable after a long period of time, then risk-management is necessary. The last most important case to consider is the large impact of the quantum threat on widely deployed standardized crypto-systems and that our society depends heavily on ICT.

This chapter provided cyber security experts and risk managers in the public and private sector tools and information to act on the quantum threat in a risk-based manner. It gives the opportunity to act on the disadvantage of quantum computing in time and fully enjoy the benefits of quantum computing.

Chapter 6

Conclusions and further research

6.1 Conclusion

This thesis investigated the requirements to run Shor's algorithm to break RSA with a 2048-bit key and used the information gathered to realize methods to support risk-management regarding the quantum threat. To break RSA, Shor's algorithm factors a known $N = pq$ into two primes p and q. RSA with a 2048-bit key (RSA-2048), means that N has a size of 2048 bits and p and q both have a size which is close to 1024 bits.

Investigating the requirements to implement Shor's algorithm led to the conclusion that there are multiple ways to implement Shor's algorithm with a quantum circuit and that there are multiple optimization or design choices possible. When making these choices there is always a trade-off between the number of logical qubits and the number of logical sequential gates. This is also known as the time-space trade-off, as the number of logical sequential gates give an indication for the duration of the computation and the logical qubits indicate an amount of space the circuit occupies. For example Beauregards circuit minimizes the required logical qubits to implement Shor's algorithm. This circuit requires approximately $2m$ logical qubits and $32m^3$ logical sequential gates, where m is the bit length of the integer $N = pq$. Resulting in approximately 4099 logical qubits and approximately $275 \cdot 10^9$ sequential logical gates for $m = 2048$ bits.

Logical qubits and gates are capable of handling errors that occur. These errors occur due to decoherence, control errors, measurement errors etc. This capability of handling errors is realized by a fault-tolerant architecture. Fault-tolerant architectures use quantum-error-correction codes to encode quantum information in multiple physical qubits. This encoding does not only protect against errors but also increases the available execution time to significantly exceed the physical coherence time of the physical qubits.

Investigation of fault-tolerant architectures led to the surface-code architecture with state distillation. It is considered the most promising fault-tolerant implementation as it has a high noise threshold, only needs a 2D qubit connectivity, and has a relatively low-cost T-gate implementation. The T-gate is currently considered the most resource demanding part of implementing quantum circuits.

For a N of size 2000 bits an estimate of the required physical resources - using the surface-code architecture with state distillation and a circuit minimizing the number of T-gates - resulted in approximately 214 million physical qubits, a minimum physical single-qubit fidelity of $99,9\%$, a minimum physical two-qubit fidelity of $99,9\%$, a minimum measurement fidelity of $99,9\%$, minimum physical qubit coherence times of $1 - 10\mu s$ and minimal physical gate duration times of $10 - 100ns$. When running this implementation, Shor's algorithm would finish in approximately 26,7 hours. However to make this estimate a highly time-optimized T-gate design was assumed. This T-gate design is not published and could not be verified. Using the published design led to a significantly higher T-gate completion time, which impacts the estimated completion time of Shor's algorithm.

No public source was found that indicated that these physical requirements could be met. Hardware platforms based on superconducting qubit technologies and ion-trap qubit technologies are currently able to realize a two-qubit gate fidelities of respectively $99,4\%$ and $99,9\%$. This is needed to be able to implement a fault-tolerant architecture. Note that only the ion-trap qubit technology is able to meet the two-qubit gate fidelity as required in the estimate. However the ion-trap qubit technology currently does not meet the 2D-qubit connectivity requirement of the surface-code architecture, leaving the superconducting qubit hardware platforms as the most likely candidate for implementing the surface-code architecture.

Currently the number of available physical (superconducting) qubits is 50. It is believed that this number of qubits is required to demonstrate quantum supremacy. Quantum supremacy is referred to as the situation that the output of a quantum computer cannot longer be simulated on a classical computer. The D-Wave quantum computer also uses superconducting qubits and has currently a larger amount of qubits available. However the D-Wave quantum computers do not facilitate circuit-based quantum computation, as is required for quantum algorithms like Shor's algorithm.

We can conclude that the gap between the available physical resources and the requirements to run Shor's algorithm and break RSA-2048 is significant. This gap can be closed by progress on the supply side, represented by the physical layer, and by progress on the demand side, represented by the logical layer and fault-tolerant layer. On the demand side, progress in reducing the fault-tolerant implementation cost will significantly reduce

the gap. Also progress in designs significantly reducing the required number of T-gates will reduce the gap. On the supply side, progress in the number of physical controllable qubits and an increase in the fidelity of quantum gates will significantly reduce the gap.

The significant gap results in concluding that it is unlikely that this gap is closed in a short period of time. However this does not implies that currently no risk-management regarding the quantum threat is required, it only implies that it is likely that we have time to prepare. Preparation is necessary considering the following cases. First, RSA is not the only vulnerable algorithm. Other algorithms will require another set of physical requirements, which will result in different gaps. Second, post-quantum algorithms, which are considered a likely risk control, are still in the process of standardization. Third, a case that should be considered is that encrypted information can be collected and stored for a long time. When there is a quantum computer, this stored information can be decrypted. If the information is still considered valuable after a long period of time, then risk-management is necessary.

The last most important case to consider is the large impact of the quantum threat on widely deployed standardized crypto-systems and that our society depends heavily on ICT. The vulnerable crypto-systems are used in various types of security protocols facilitating services like online shopping and banking, online access to your health-care test results, online registration for social funds for citizens, online access to your kid's daycare reports or student files, providing to citizens online trusted information about calamities. Not being able to trust these technologies supporting these and other online services cripples society. To prevent this impact on society preparations need to be start in time, this requires risk-management.

The national cyber security center (NCSC) is already using its role as expert authority on cyber security to provide public and private parties information about the quantum threat. If the NCSC starts monitoring the quantum threat, then they are able to monitor if risks can be and are mitigated fast enough to prevent a large impact on society. It is also important for the government to connect to international strategies as our society flourishes with an international inter-operable ICT landscape. This fits with the ambition -as formulated in the second national cyber security strategy (NCSS2)- to play a prominent role in reaching internationally accepted standards related to actions in the digital domain.

To monitor the quantum-treat the NCSC can use the proposed monitoring framework, which includes the factors reducing the gap between the demand side and the supply side. By changing the focus from Shor's algorithm to a more general formulation including all algorithms which impact cyber security and adding the topic user-friendliness to estimate the threat-actor's required capabilities, the demand and supply side of the monitoring

framework are completed. The monitoring categories post-quantum cryptography and quantum cryptography are added to the monitoring framework as risk controls. These two groups of technologies can be applied to mitigate to a quantum-safe infrastructure. Monitoring these items is necessary as standardization is in progress.

Not only the government has its responsibilities regarding cyber security but also organizations as described in NCSS2. This also defines a role for organizations in the public and the private sector regarding the quantum threat. To make the quantum-treat more concrete for organizations a method for translating the threat was proposed. It enables to translate the technically formulated impact to how it impacts the strategic risks of an organization. Additionally this method was explained by an example, using non-compliance to the general data protection regulation (GDPR) as strategic risk.

Depending on the significance of the impact on strategic risks, actions can be delayed or need to be taken. In the proposed *pragmatic approach* three scenarios are described. Each scenario has guidelines for actions depending on the impact and its significance on strategic risks. Using this pragmatic approach creates a balance between the uncertainty about when the risk materializes and the investments required to investigate and act upon the risk. This is different compared to other approaches, which use an asset-based approach and start prioritizing after a full inventory of vulnerable assets.

The monitoring framework, the translation method and the pragmatic approach give the public and private sector tools to act on the quantum-threat. This is one step in enabling society to reduce the negative consequences of quantum computing on society and to fully benefit from positive consequences: the promised progress in research fields like medicine, material science, logistics and energy.

6.2 Recommendation for further research

To further investigate when a large-scale quantum computer is available to pose a threat to cyber security the following research subjects can be formulated:

- How to implement a surface-code like architecture on ion-trap qubit technologies?
- Investigate the physical requirements for breaking other algorithms e.g. DSA.
- Can alternatives to circuit-based-quantum computing effect cyber security?
- Improving fault-tolerant architectures.
- The impact of compilers for quantum computers on the quantum-threat.

Results from these subjects contribute to the topics of the monitoring framework as proposed in the Chapter 5. It would also be interesting to collect historical data on this

threat, by storing each update for the identified monitoring topics. This way it could be validated if the monitoring topics were sufficient to keep control of the threat. If this approach turns out to be successful, then the method for determining the monitoring topics could be applied for new threats, which are unclear on how these evolve in time.

The pragmatic approach should be validated by conducting a case study. This is also true for the translation method. A case study can quantify the investments required of both approaches, the pragmatic approach and Mosca & Mulhollands Quantum Risk Assessment (QRA).

- Validating the pragmatic approach by performing a case study.
- Validating the translation method by performing a case study.
- Comparing the pragmatic approach with the QRA methodology in a case study.

The impact of quantum computing on society can be investigated further. Research questions on that topic can be.

- Supports the formulation of the protection requirement in the General Data Protection Regulation (GDPR) early mitigation regarding the quantum-threat?
- Is quantum computation capable of changing the international relations between countries?

This thesis focused on the negative effects of a quantum on cyber security, also opportunities for cyber security can be identified e.g.,

- Can cyber security profit from small-scale quantum computers?
- How to certify quantum cryptography?
- New post-quantum algorithms.
- Quantum networks.

Appendix A

Example: Computing the period r

Example of computing the periodicity of a function

For $x = 2$ and $N = 15$, compute the sequence: $1 = x^1 \mod N, x^2 \mod N, \ldots$, until the results show a cycle.

$$
\begin{aligned}
2^0 &\mod 15 = 1,\\
2^1 &\mod 15 = 2,\\
2^2 &\mod 15 = 4,\\
2^3 &\mod 15 = 8,\\
2^4 &\mod 15 = 1,\\
2^5 &\mod 15 = 2,\\
2^6 &\mod 15 = 4,\\
2^7 &\mod 15 = 8,\\
2^8 &\mod 15 = 1,
\end{aligned}
$$

The results start to repeat after $2^4 \mod 15 = 1$ and after $2^8 \mod 15 = 1$ etc. The smallest period r is 4, for which holds $x^r = 1 \mod N$.

This example is derived from the example given in [46].

Appendix B

Positive interference using the Quantum Fourier Transform

The QFT applied to state $|l\rangle$ is defined as: $F_N\left(|l\rangle\right) = \frac{1}{\sqrt{N}} \sum_{y=0}^{N-1} e^{2\pi i(yl/N)}|y\rangle$, where $N = 2^n$. For a quantum state $\frac{1}{\sqrt{m}} \sum_{j=0}^{m-1} |j\rangle$ the QFT is formulated as:

$$F_N\left(\frac{1}{\sqrt{m}} \sum_{j=0}^{m-1} |j\rangle\right) = \frac{1}{\sqrt{m}} \sum_{j=0}^{m-1} \frac{1}{\sqrt{2^n}} \sum_{y=0}^{2^n-1} e^{2\pi i(yj/2^n)}|y\rangle$$

We like to find the hidden period of state: $\frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle|a^x \mod N\rangle$. Before we start with the calculations we make some observations, these are derived from [13].

Observation 1: Note that if we observe the second register, some value f(s) is obtained, with $s < r$. Note that $f(x) = f(s)$, with $f(x) = |a^x \mod N\rangle$ if and only if $x = s \mod r$

Observation 2: Note that multiple elements can have the value $f(s)$. Let m be the number of elements of $\{0, \ldots, 2^n - 1\}$ that map to the observed value of $f(s)$. Because $x = s \mod r$, the x are of the from $x = s + jr$, for $0 \leq j < m$ (Remember these are the x for which $f(x) = f(s)$ holds).

Observation 3: When observing the second register, it collapses to the classical value of $f(s)$, while the first register will collapse[1] to a superposition of $|s\rangle$, $|s + r\rangle$, $|s + 2r\rangle$, $|s + 3r\rangle$, ..., $|s + (m - 1)r\rangle$. Recall that m is the number of elements that map to the observed value $f(s)$, we can write the first register as: $\frac{1}{\sqrt{m}} \sum_{j=0}^{m-1} |s + jr\rangle$.

[1] The first register also collapses due to the entanglement.

Apply the QFT and the observation that x is of the from $x = s + jr$, for $0 \leq j < m - 1$, this results in:

$$\frac{1}{\sqrt{m}} \sum_{j=0}^{m-1} \frac{1}{\sqrt{2^n}} \sum_{y=0}^{2^n-1} e^{2\pi i \left(y\left(\frac{s+jr}{2^n}\right)\right)} |y\rangle$$

Apply the rule $e^{a+b} = e^a \cdot e^b$ and some rearrangement gives:

$$\frac{1}{\sqrt{m}} \sum_{j=0}^{m-1} \frac{1}{\sqrt{2^n}} \sum_{y=0}^{2^n-1} e^{2\pi i \left(y\left(\frac{s+jr}{2^n}\right)\right)} |y\rangle = \frac{1}{\sqrt{m \cdot 2^n}} \sum_{y=0}^{2^n-1} e^{2\pi i \left(\frac{sy}{2^n}\right)} \sum_{j=0}^{m-1} e^{2\pi i \left(\frac{jry}{2^n}\right)} |y\rangle$$

Now we like to check for which $|y\rangle$, constructive interference will occur. First we need to apply (the proof can be found in Appendix B of [13]):

$$\sum_{j=0}^{m-1} z^j = \begin{cases} \frac{1-z^m}{1-z} & \text{for } z \neq 1 \\ m & \text{for } z = 1 \end{cases}$$

This results in:

$$\sum_{j=0}^{m-1} \left(e^{2\pi i \left(\frac{ry}{2^n}\right)}\right)^j = \begin{cases} \frac{1-e^{2\pi i (ry/2^n) m}}{1-e^{2\pi i (ry/2^n)}} & \text{for } e^{2\pi i \left(\frac{ry}{2^n}\right)} \neq 1 \\ m & \text{for } e^{2\pi i \left(\frac{ry}{2^n}\right)} = 1 \end{cases}$$

For simplicity we assume that r divides 2^n, if this is not true another proof can be found in [13] page 31-32. If r divides 2^n, the r fits an integer number of times in the domain $\{0, \ldots, 2^n - 1\}$ of f and $m = 2^n/r$. Note that $e^{2\pi i (ry/2^n)} = 1$ is only true if and only if $ry/2^n$ is an integer if and only if y is multiple of $2^n/r$. If this holds for the y, then measuring the quantum state will give $(|\frac{m}{\sqrt{m \cdot 2^n}}|^2)$, will result in obtaining the y for which hold $y = c \cdot \frac{2^n}{r}$ with a high probability.

Recall $1 < m \leq 2^n - 1$, for n is large, this will result not in a high probability. However if $y = c \cdot \frac{2^n}{r}$, then the solution for $e^{2\pi i \left(\frac{ry}{2^n}\right)} \neq 1$, $\frac{1-e^{2\pi i (ry/2^n) m}}{1-e^{2\pi i (ry/2^n)}}$, results in zero. The probability of measuring an y, that does not lead to a period r is zero.

For the difficult case, as described in [13], this does not hold. There is a small probability of measuring a y that does not solve to a useful r. Hence the note that Shor's algorithm is probabilistic.

Appendix C

Background information for the surface-code architecture

Recall that operators are complex valued $M \times M$ matrices. The operators X, Y, I and Z are defined as $X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ $Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$, $I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$, and $Y = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$.

Operators

The measurement M_z returns eigenvalues $+1$ and projects to eigenstate $|g\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ or eigenvalue -1 and project to eigenstate $|e\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$.

The measurement M_x returns eigenvalues $+1$ and projects to eigenstate $|+\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ or eigenvalue -1 and project to eigenstate $|-\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix}$.

The Hadamard operation $H = \frac{1}{\sqrt{2}} (X + Z) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$.

The S-gate is defined as $S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$.

The T-gate is defined as $T = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}$

The CNOT gate, the controlled-not gate is defined as

$$CNOT = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}.$$

This is a two qubit gate, where the first state is the control qubit and the second state the target.

The Toffoli gate, the controlled-controlled-not (CCNOT) gate is defined as

$$CCNOT = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}.$$

The first two states are the controls and the third is the target.

Commuting and Anti-commuting

Two operators commute if $XZ = ZX$ holds. Two operators, X and Z anti-commute if $XZ = -ZX$ holds. The last mathematical property can be compared to the physical property that e.g. amplitude and phase cannot be simultaneously measured [10].

Qubit properties

Any two-level quantum system that satisfies the relations below can be used as a qubit [10].

- $X^2 = Z^2 = Y^2 = I$,
- $XZ = -ZX$,
- $[X, Y] = XY - YX = -2Z$.

Note that also $[Y, Z] = -2X$ and $[Z, X] = +2Y$.

Simple non-destructive quantum-error detection

A non-destructive quantum-error detection can be realized by measuring multiple qubits simultaneously with measurement operators that commute [10]. Consider a two-qubit system, with qubit a and qubit b. This system is measured using the two-qubit operator X_aX_b and Z_aZ_b. These operators represent separate M_x and M_z measurements, but the do commute:

$$\begin{aligned}[X_aX_b, Z_aZ_b] &= (X_aX_b)(Z_aZ_b) - (Z_aZ_b)(X_aX_b) \\ &= X_aZ_aX_bZ_b - Z_aX_aZ_bX_b \\ &= (-Z_aX_a)(-Z_bX_b) - Z_aX_aZ_bX_b = 0.\end{aligned}$$

Using these measurement operators and the four Bell states will create a non-destructive error-detecting property. The measurement will project the quantum state onto one of the other two-qubit eigenstates (Bell states), see Table C.1. If the measurement eigenvalue is changed compared to the previous result, then a error has occurred. To uniquely identify errors a more complex system is needed, which is provided by the surface-code architecture [10].

Z_aZ_b	X_aX_b	**Eigenstates**
$+1$	$+1$	$\|\phi_1\rangle = \frac{1}{\sqrt{2}}(\|gg\rangle + \|ee\rangle)$
$+1$	-1	$\|\phi_2\rangle = \frac{1}{\sqrt{2}}(\|gg\rangle - \|ee\rangle)$
-1	$+1$	$\|\phi_3\rangle = \frac{1}{\sqrt{2}}(\|ge\rangle + \|eg\rangle)$
-1	$+1$	$\|\phi_2\rangle = \frac{1}{\sqrt{2}}(\|ge\rangle - \|eg\rangle)$

TABLE C.1: Eigenstates of the two-qubit operators X_aX_b and Z_aZ_b and the four eigenstates (Bell states) for this example. Derived from [10].

Appendix D

Brief overview of commercially quantum computers

In the last chapter we have seen that there are multiple options to create the hardware platform for quantum computers. It is still to be determined which technology, or which two technologies, will eventually become mainstream for quantum computing. This does not only lead to diversity in technologies applied in academic research labs but also to diversity in commercial available quantum computers. In order to track the progress of these commercial available quantum computers, this chapter provides a brief overview these systems.

D.1 IBM

Currently IBM offers a 16 qubit processor and has the ambition to build IBM Q systems with ~ 50 qubits in the next few years. IBM also focuses on developing new applications. To facilitate this development two main approaches are defined. First IBM collaborates with third parties from the commercial sector and second IBM developed a environment which enables quantum computing in the cloud. This environment facilitates a open research community to develop quantum algorithms [36].

For the quantum algorithm research IBM provides a compiler. This compiler maps the desired algorithm, written with Python[1], to the available hardware [36]. This increases the user-friendliness of using a quantum computer and enables the search to applications requiring less physical qubits than Shor's algorithm.

[1]The compiler tutorial can be found
https : *//github.com/QISKit/qiskit−tutorial/blob/master/*1*_introduction/compiling_and_running.ipynb*

D.2 Rigetti

Rigetti provides two types of services. Simulation up to 30 qubits on a classical computer and a real quantum processor of 8 superconducting qubits. This quantum processor is part of Rigetti's Forest 1.0, which allows users to program and run hybrid algorithms combining classical and quantum computations within an algorithm [47].

D.3 Intel

In January 2018 Intel released their 49 superconducting qubit test chip to QuTech [37]. Intel also contributes to other quantum technologies, like spin qubits [48].

D.4 Microsoft

Microsoft's ambition is to build a topological quantum computer, that is a quantum computer built of topological qubits. These topological qubits should in theory be more robust to noise, which results in higher coherence times. This will reduces the requirements for error correction and the number of physical qubits needed to build a single logical qubit [49].

Topological qubits are created using the topological properties of non-abelian anyons. This is a group of quasiparticle, particle-like objects that emerges from the interactions inside matter. The information is encoded in the order in which we swap the positions of the anyons. This swapping is called braiding, because the pattern of swaps, between neighbouring pairs of anyons, through space and time look braided [50].

However only the simplest species of anyons is observed, more complex non-abelian anyons are yet to be observed. The simplest species is observed in 2012 by L. Kouwenhoven at the Delft University of Technology in the Netherlands [50].

Microsoft is also doing research in the field of quantum applications and uses classical computers to simulate quantum circuits.

D.5 D-Wave

D-wave is known as the first company that made a quantum computer available. However the quantum mechanical speed up is for the D-Wave quantum computer still under debate [34]. The behaviour of the D-Wave quantum computer is consistent with quantum annealing. This will be elaborated on in the next section.

D.5.1 Adiabatic quantum computing

In this thesis we focused on digital or circuit-based quantum computing. However, there is another branch of quantum computing, this is analog or adiabatic quantum computing. Adiabatic quantum computers are mainly used for optimization problems: find a set of variables that minimizes a multi-variable function [31].

For adiabatic quantum computing the problem to solve is directly constructed in the physical system Hamiltonian. The solution follows from measuring the resulting ground state (lowest energy state) [51]. The value of the gap between the first excited state and the ground state is not well understood. Understanding this is necessary to determine the speed with which the parameters can be changed to turn the system into its end state [31]. If the speed is incorrect, the result measured will be useless.

To ensure a good operation an adiabatic quantum computer needs extremely low temperatures. A quantum annealing machine works similar to an adiabatic quantum computer, it however can operate at a more practical temperature level [31]. The D-Wave quantum computer is of the type quantum annealing [34] and uses superconducting qubits. However it is not clear how to perform error correction on an adiabatic type of quantum computer[2].

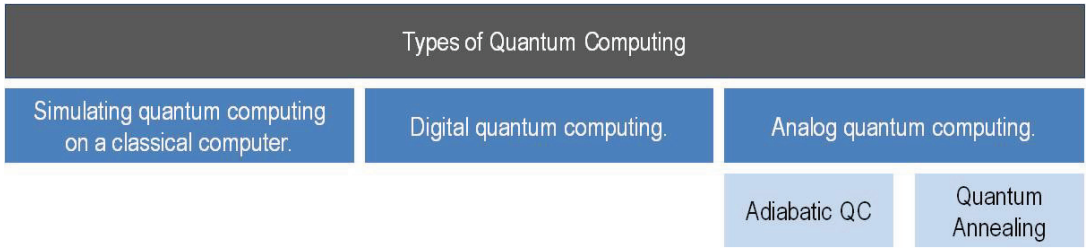


FIGURE D.1: Types of Quantum Computing.

D.6 Google

Google Quantum Artificial Intelligence Lab studies how quantum computing might advance machine learning. The lab houses a quantum computer from D-Wave [52]. Google is also building a chip of 49 superconducting qubits [38] and they develop a prototype that combines the adiabatic quantum computing method with the digital approachs error-correction capabilities [53].

[2]Note: A quantum computer using superconducting qubit technology does not have to be a circuit base quantum computer, and therefore is not per definition able to run an algorithm like Shor's.

Bibliography

[1] Rodney Doyle Van Meter III. *Architecture of a Quantum Multicomputer Optimized for Shors Factoring Algorithm*. PhD thesis, February 2008. https://arxiv.org/abs/quant-ph/0607065v1.

[2] M. A. Nielsen and I.L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2007.

[3] L. Kouwenhoven. Quantumcomputer, October 2016. https://tvblik.nl/de-universiteit-van-nederland/leo-kouwenhoven-quantumcomputer.

[4] Retrieved 24 October 2017 https://www.rigetti.com/.

[5] L. Chen, S. Jordan, Y. Liu, D. Moody, R. Peralta, R. Perlner, and D. Smith-Tone. Report on post-quantum cryptography. Technical report, National Institute of Standards and Technology, April 2016. URL http://dx.doi.org/10.6028/NIST.IR.8105.

[6] M. Campagna, L. Chen, Ö. Dagdelen, J. Ding, J. Fernick, N. Gisin, D. Hayford, T. Jennewein, N. Ltkenhaus, M. Mosca, B. Neill, M. Pecen, R. Perlner, G. Ribordy, J. Schanck, D. Stebila, N. Walenta, W. Whyte, and Z. Zhang. Quantum safe cryptography and security an introduction, benefits, enablers and challenges. Technical report, European Telecommunications Standards Institute, June 2015. URL http://www.etsi.org/images/files/ETSIWhitePapers/QuantumSafeWhitepaper.pdf.

[7] National Cyber Security Center. Postkwantumcryptografie - bescherm uw data vandaag tegen de dreiging van morgen. Technical report, Ministry of Justice and Safety, August 2017. URL https://www.ncsc.nl/actueel/factsheets/factsheet-postkwantumcryptografie.html.

[8] M. Amy, O. Di Matteo, V. Gheorghiu, M. Mosca, A. Parent, and J. Schanck. Estimating the cost of generic quantum pre-image attacks on sha-2 and sha-3. Cryptology ePrint Archive, 2016. Retrieved 12 October 2017 http://eprint.iacr.org/2016/992.

[9] E. Barker and A. Roginsky. Transitions: Recommendation for transitioning the use of cryptographic algorithms and key lengths. Technical report, National Institute of Standards and Technology, November 2015. URL http://dx.doi.org/10.6028/NIST.SP.800-131Ar1.

[10] A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland. Surface codes: Towards practical large-scale quantum computation. *A Physical Review*, 86(032324), september 2012.

[11] T. Simonite. Quantum computing paranoia creates a new industry. Article in MIT Technology Review, in the category Connectivity, January 2017. https://www.technologyreview.com/s/603424/quantum-computing-paranoia-creates-a-new-industry.

[12] V. Gheorghiu. Quantum resource estimation. Presentation at ETSI/IQC Quantum-Safe Workshop, London UK, Sep 14, 2017, september 2017.

[13] R. de Wolf. Quantum computing: Lecture notes. Technical report, CWI, May 2016. URL http://homepages.cwi.nl/~rdewolf/qcnotes.pdf.

[14] D. Beckman, A. Chari, S. Devabhaktuni, and J. Preskill. Efficient networks for quantum factoring. Phys. Rev.A54:1034-1063,1996, 1966. https://arxiv.org/abs/quant-ph/9602016v1.

[15] S. Lloyd. Quantum information science. Technical report, Massachusetts Institute of Technology, 2009. URL http://web.mit.edu/2.111/www/notes09/spring.pdf.

[16] V. Vedral, A. Barenco, and A. Ekert. Quantum networks for elementary arithmetic operations. *Phys. Rev. A*, 54:147–153, July 1996. Retrieved via https://arxiv.org/abs/quant-ph/9511018v1.

[17] T. D. Ladd, F. Jelezko, R. Laflamme, Y. Nakamura, C. Monroe, and J. L. OBrien. Quantum computing. *arXiv:1009.2267v1*, June 2009. URL https://arxiv.org/abs/1009.2267.

[18] B. Terhal E. Campbell and C. Vuillot. Roads towards fault-tolerant universal quantum computation. *Nature*, 549, September 2017. URL http://www.nature.com/nature/journal/v549/n7671/full/nature23460.html.

[19] S. Devitt, W. Munro, and K. Nemoto. Quantum error correction for beginners. *arXiv:quant-ph/0905.2794v4*, June 2013. URL https://arxiv.org/abs/0905.2794.

[20] J. Held. Keynote: Engineering a scalable quantum computer architecture. Keynote at First International Workshop on Quantum Computer Architecture - slides not published yet, November 2017. 30 November 2017 https://qutech.nl/events/international-workshop-on-quantum-computer-architecture/.

[21] P. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26:14841509, 1997. Retrieved via https://arxiv.org/abs/quant-ph/9508027v2.

[22] M. Mosca and J. Mulholland. A methodology for quantum risk assessment. Published on the site of the Global Risk Institute, January 2017. Retrieved September 2017 http://globalriskinstitute.org/publications/3423-2/.

[23] M. Mosca. A quantum of prevention for our cybersecurity. Published on the site of the Global Risk Institute with the title Quantum Computing: A New Threat to Cybersecurity, September 2016. Retrieved 4 September 2017 http://globalriskinstitute.org/publications/quantum-computing-cybersecurity/.

[24] M. Mosca. Cybersecurity in an era with quantum computers: will we be ready? Cryptology ePrint Archive, 2015. Retrieved 4 September 2017 https://eprint.iacr.org/2015/1075.pdf.

[25] L. Vandersypen, M. Steffen, G. Breyta, C. Yannoni, M. Sherwood, and I. Chuang. Experimental realization of shor's quantum factoring algorithm using nuclear magnetic resonance. *Nature*, 414:883–887, December 2001.

[26] J. Vartiainen, A. Niskanen, M. Nakahara, and M. Salomaa. Implementing shors algorithm on josephson charge qubits. *PHYSICAL REVIEW A*, 70, July 2004.

[27] R. van Meter and C. Horsman. A blueprint for building a quantum computer. *Communications of the ACM*, 56(10):84–93, Oktober 2013.

[28] Simon J. Devitt, Austin Fowler, and Lloyd C. L. Hollenberg. Investigating the practical implementation of shor's algorithm. *Proceedings of SPIE - The International Society for Optical Engineering*, 5650, 02 2005.

[29] R. Versluis. Interview versluis. not published, October 2017.

[30] X.Fu, L.Riesebos, L.Lao, C.G.Almudever, F.Sebastiano, R.Versluis, E.Charbon, and K.Bertels. A heterogeneous quantum computer architecture. pages 323–330, May 2016. URL https://dl.acm.org/citation.cfm?id=2906827.

[31] A. Fruchtman and I. Choi. Technical roadmap for fault-tolerant quantum computing. Technical report, Networked Quantum Information Technologies - University of Oxford, October 2016. URL https://nqit.ox.ac.uk/content/technical-roadmap-fault-tolerant-quantum-computing.

[32] W. K. Hensinger. Constructing a practical quantum computer. Devoxx-YouTube, May 2017. https://www.youtube.com/watch?v=uHw6ikgknXQ.

[33] S. Weidt, J. Randall, S. Webster, K. Lake, A. Webb, I. Cohen, T. Navickas, B. Lekitsch, A. Retzker, and W. Hensinger. Trapped-ion quantum logic with global radiation fields. *PHYSICAL REVIEW LETTERS*, 117(220501):1–6, November 2016.

[34] G. Wendin. Quantum information processing with superconducting circuits: a review. https://arxiv.org/abs/1610.02208, October 2017.

[35] M. Devoret and R. Schoelkopf. Superconducting circuits for quantum information: An outlook. *SCIENCE*, 339:1–6, March 2013. Retrieved via http://science.sciencemag.org/content/339/6124/1169/tab-pdf.

[36] C. Vu. IBM announces advances to IBM quantum systems and ecosystem. IBM New room, November 2017. Retrieved on 5 December 2017 via http://www-03.ibm.com/press/us/en/pressrelease/53374.wss.

[37] J. Hsu. Ces 2018: Intel's 49-qubit chip shoots for quantum supremacy. spectrum.ieee.org, January 2018. Retrieved on 17 January 2018 via https://spectrum.ieee.org/tech-talk/computing/hardware/intels-49qubit-chip-aims-for-quantum-supremacy.

[38] R. Courtland. Google plans to demonstrate the supremacy of quantum computing. spectrum.ieee.org, May 2017. Retrieved on 17 January 2018 via https://spectrum.ieee.org/computing/hardware/google-plans-to-demonstrate-the-supremacy-of-quantum-computing.

[39] Quantum technology beginning come its own. The McKinsey picture was published in essay on www.economist.com, March 2017. The picture was retrieved from http://www.economist.com/news/essays/21717782-quantum-technology-beginning-come-its-own.

[40] Directorate-General Justice and Consumers. Protection of personal data. European Commission archive, Justice section, data protection page, December 2016. Retrieved on 7 December 2017 via http://ec.europa.eu/justice/data-protection/.

[41] Monitoring Commissie Corporate Governance Code. De nederlandse corporate governance code. Website Monitoring Commissie Corporate Governance Code, December 2016. Retrieved on 7 December 2017 via http://www.mccg.nl/nieuws/5720/Code-2016-wettelijk-verankerd.

[42] National coordinator for Security and counterterrorism in The Netherlands. National cyber security strategy 2. URL https://www.ncsc.nl/english/current-topics/national-cyber-security-strategy.html.

[43] NIST. Post-quantum cryptography - workshops and timeline. csrc.nist.gov, December 2017. Retrieved on 03 Januari 2018 via https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/Workshops-and-Timeline.

[44] Product site idq. Product site - id Quantique. Retrieved on Januari 3, 2018 via https://www.idquantique.com/quantum-safe-security/products/cerberis-qkd-blade/.

[45] Sheng-Kai Liao, Wen-Qi Cai, Liang Zhang Wei-Yue Liu and, Yang Li, Ji-Gang Ren, Juan Yin, Qi Shen, Yuan Cao, Zheng-Ping Li, Feng-Zhi Li, Xia-Wei Chen, Li-Hua Sun, Jian-Jun Jia, Jin-Cai Wu, Xiao-Jun Jiang, Jian-Feng Wang, Yong-Mei Huang, Qiang Wang, Yi-Lin Zhou, Lei Deng, Tao Xi, Lu Ma, Tai Hu, Qiang Zhang, Yu-Ao Chen, Nai-Le Liu, Xiang-Bin Wang, Zhen-Cai Zhu, Chao-Yang Lu, Rong Shu, Cheng-Zhi Peng, Jian-Yu Wang, and Jian-Wei Pan. Satellite-to-ground quantum key distribution. *Nature*, 549, September 2017. URL https://www.nature.com/articles/nature23655.

[46] Hacking at quantum speed with shor's algorithm — infinite series. YouTube, 2017. https://www.youtube.com/watch?v=wUwZZaI5u0c.

[47] C. Rigetti. Introducing forest 1.0. Medium.com site, June 2017. Retrieved 24 October 2017 https://medium.com/rigetti/introducing-forest-f2c806537c6d.

[48] Intel PR. Intel delivers 17-qubit superconducting chip with advanced packaging to qutech. newsroom.intel.com, October 2017. Retrieved on 28 December 2017 via https://newsroom.intel.com/news/intel-delivers-17-qubit-superconducting-chip-advanced-packaging-qutech/.

[49] A. Linn. microsoft doubles quantum computing bet. The AI Blog - the Official Microsoft Blog, November 2016. Retrieved 12 October 2017 at 13:34 (UTC+1) on https://blogs.microsoft.com/ai/2016/11/20/microsoft-doubles-quantum-computing-bet/.

[50] E. Gibney. Inside microsofts quest for a topological quantum computer. Nature.com - News site, October 2016. Retrieved 12 October 2017 on http://www.nature.com/

news/inside-microsoft-s-quest-for-a-topological-quantum-computer-1.20774.

[51] R. Naus. Interview naus. not published, November 2017.

[52] H. Neven. Launching the quantum artificial intelligence lab. research.googleblog.com, May 2013. Retrieved on 17 January 2018 via https://research.googleblog.com/2013/05/launching-quantum-artificial.html.

[53] P. Ball. Google moves closer to a universal quantum computer. Nature News website, June 2016. Retrieved on 12 October 2017 via http://www.nature.com/news/google-moves-closer-to-a-universal-quantum-computer-1.20032.