UCL DEPARTMENT OF ELECTRONIC
AND ELECTRICAL ENGINEERING

Quantum Logic Circuits for Solid-State Quantum Information Processing

Andrea Del Duce

A thesis submitted to University College London for the degree of Doctor of Philosophy (Ph.D)

Department of Electronic and Electrical Engineering

University College London

October 2009

I, Andrea Del Duce, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the thesis.

Abstract

This thesis describes research on the design of quantum logic circuits suitable for the experimental demonstration of a three-qubit quantum computation prototype. The design is based on a proposal for optically controlled, solid-state quantum logic gates. In this proposal, typically referred to as SFG model, the qubits are stored in the electron spin of donors in a solid-state substrate while the interactions between them are mediated through the optical excitation of control particles placed in their proximity.

After a brief introduction to the area of quantum information processing, the basics of quantum information theory required for the understanding of the thesis work are introduced. Then, the literature on existing quantum computation proposals and experimental implementations of quantum computational systems is analysed to identify the main challenges of experimental quantum computation and typical system parameters of quantum computation prototypes. The details of the SFG model are subsequently described and the entangling characteristics of SFG two-qubit quantum gates are analysed by means of a geometrical approach, in order to understand what entangling gates would be available when designing circuits based on this proposal. Two numerical tools have been developed in the course of the research. These are a quantum logic simulator and an automated quantum circuit design algorithm based on a genetic programming approach. Both of these are used to design quantum logic circuits compatible with the SFG model for a three-qubit Deutsch-Jozsa algorithm. One of the design aims is to realise the shortest possible circuits in order to reduce the possibility of errors accumulating during computation, and different design procedures which have been tested are presented. The tolerance to perturbations of one of the designed circuits is then analysed by evaluating its performance under increasing fluctuations on some of the parameters relevant in the dynamics of SFG gates. Because interactions in SFG two-qubit quantum gates are mediated by the optical excitation of the control particles, the solutions for the generation of the optical control signal required for the proposed quantum circuits are discussed. Finally, the conclusions of this work are presented and areas for further research are identified.

Acknowledgements

The work presented in this thesis would not have been possible without the help of many people to whom I am deeply indebted. First of all, I would like to thank my first supervisor Prof. Polina Bayvel. Her support and advice always gave me the encouragement needed along the journey and I always appreciated her guidance both as a supervisor and a friend. I am also grateful for giving me the opportunity to join the Optical Networks Group and UCL, which I found to be an incredibly inspiring working environment.

I would like to thank my second supervisor Dr. Tony Harker for his ongoing support and for always finding a way of explaining many (to me) obscure physical concepts in a simple and clear manner. I am also very grateful to Prof. Marshall Stoneham. His feedback and suggestions really helped me to shape and focus my research. Thanks are also due to Dr. Thornton Greenland for all the time he dedicated to introducing me to the project and for many discussions along the way, for which I am also grateful to Prof. Andrew Fisher.

I would like to thank Dr. Paul Radmore and Dr. Seb Savory for their invaluable help with the mathematics concerning the problems addressed in Chapter 4. I also owe my gratitude to Dr. Benn Thomsen who, the few times in which my work brought me to the lab, guided and helped me with patience despite my complete inexperience.

I would not have enjoyed my work so much without the company of all the colleagues from room 808. In the years many have left and many have come, and all have contributed to make to hours in the office happy and stimulating.

I owe special gratitude to a number of people inside and outside UCL who have made my time in London a special one. Gianca and Elena, who have been my family in London for many years and I will never forget the beautiful time I had with them (as well as Elena's fantastic cooking!). Lorenzo, our friendship started through basketball and movies. Maybe we are getting too old for basketball, but at least we still have endless movies to discuss! The three people sitting closest to me in the office are responsible for a lot of my good time at work: Yannis (sitting in front of me, with whom we exchanged many troubled looks when simulations were returning mysterious

results), Ben (sitting next to me who, apart from stealing my pens and invading my desk, introduced me to Indie rock, tea and his strange rules on humour) and Chin-Pang (sitting behind me, who is wise and very often knows the correct answer). I am also very grateful to Costanza, who has been an excellent flatmate and a great friend.

I would like to thank my parents for everything they have done for me and for always supporting me in everything I choose to do.

Finally, I can never thank enough my girlfriend Fede who has been my source of happiness and strength throughout the years.

Contents

List of tables

List of figures

Chapter 1 Introduction

A quantum computer is a computational system based on the interaction of two-level quantum mechanical systems called quantum bits or, as first introduced in [Sch95], qubits. It is potentially very powerful due to the combination of the three following features [Nie03]. Firstly, thanks to the superposition of quantum states, a quantum register comprising N qubits can simultaneously be loaded with 2^N different values which can then be processed in parallel. Thus, a quantum computer naturally implements parallel computation. Secondly, the values stored in a quantum register can be made to interfere, a technique which can be used to extract out of all superimposed values the correct solution of an algorithm. Thirdly, in a quantum computer it is possible to introduce a special correlation, called *entanglement*, between the qubits which allows one to influence the state of one qubit by operating on another, a phenomenon that has no equivalent in classical computation.

There are different approaches by which the idea of a computational system based on quantum mechanics was developed. Feynman, for example, observed that simulating large quantum mechanical systems on classical computers was computationally a difficult problem to solve due to the considerable resources it requires and argued that these difficulties may be circumvented if the computational machine was itself based on quantum mechanics[Fey82]. Others suggested the need for new computational models analysing Moore’s law which states that computational power doubles for constant cost roughly once every two years. It has been observed, that approximately around the year 2020 the increase in computational power described by Moore will come to an end when miniaturisation will reach levels in which quantum effects will interfere with the functioning of electronic devices and that the solution to this problem could come from switching to quantum computational models [Nie03]. However, it was probably the theoretical framework set by Deutsch in his studies on the potentials of quantum computers[Deu85] which mostly shaped the modern concept of quantum computation[Nie03]. Nevertheless, although these first studies suggested that quantum computers could lead to increased computational power compared to classical systems, no groundbreaking application could be found until 1994 when Shor demonstrated in [Sho94] that a quantum computer would have been able to factorize large integers exponentially faster than a classical machine thereby posing a serious threat to a widely used cryptographic system termed RSA (which takes its name from the first letters of the surnames of its inventors Rivest, Shamir and Adleman). The relation between Shor's quantum factoring algorithm and the RSA cryptographic system is the following: As reviewed, for example, in [Nie03], the fastest classical factorisation algorithms require a number of operations which grows exponentially with the size of the problem. The factorization problem is therefore considered to be intractable on a classical computer since, for sufficiently long integers, the computation time becomes excessively long. In 1994, for example, it took 8 months to factor a 129 digit integer and while today a shorter time would be necessary for a similar number, computational times are still inaccessible for integers larger than 600 digits [Ger05]. This difficulty in calculating the factors of large integers is exploited for obtaining encryption keys in the RSA encryption system. However, Shor's quantum factorization algorithm only requires a number of operations which grows polynomially with the size of the problem[Nie03,Ger05] thereby showing that if it were possible to build a quantum computer able to process integers in the order of hundreds of digits then the RSA system would no longer be safe [Sho94]. Figure 1-1 shows the number of operations required for factorizing large integers for increasing size of the number to be factored, comparing the classical factorization algorithm (exponential growth) to the quantum case (polynomial growth)[Nie03,Ger05].

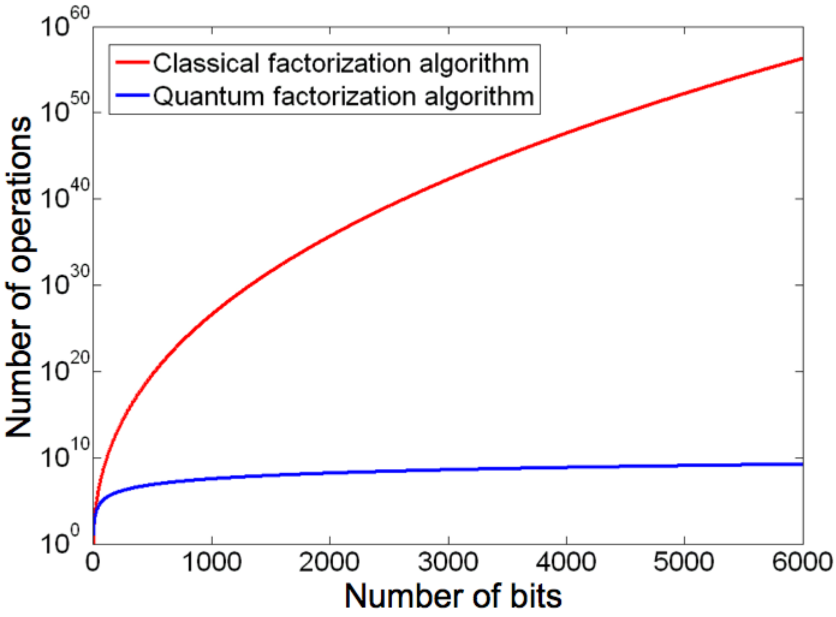


Figure 1-1: Number of operations necessary for factorizing integers with a classical factorization algorithm compared to the quantum case. The red line shows the exponential growth of the classical factorization algorithm as opposed to the polynomial growth of the quantum algorithm for increasing size of the number of bits corresponding to the integer to be factored[Nie03,Ger05].

The implications of Shor’s discovery on communication security really triggered the interest and research in the area of quantum computation on a large scale, leading, over the years, to substantial theoretical and experimental breakthroughs. After Shor’s factoring algorithm, other quantum algorithms were discovered which are able to solve some specific problems more efficiently than classical computers. Amongst these, the quantum search algorithm demonstrated by Grover[Gro97] attracted great interest. Whilst the search for a specific object in an unsorted space of L elements through a classical approach requires a number of steps proportional to L, Grover showed that the same problem could be solved in a number of steps proportional to $\sqrt{L}$ using a quantum computer. Although this speed-up is less impressive compared to the one obtained in Shor’s factorization algorithm, there are a wide range of classical algorithms which rely on search strategies which would benefit from the quantum approach[Nie03]. Finally, also Feynman’s idea of efficiently modelling quantum mechanical systems through quantum computers rather than classical machines was further developed[Llo96] in the light of the new achievements.

Other theoretical studies focused on how to correctly implement a quantum computational system despite the unavoidable presence of errors deriving from, both, the unwanted coupling of the environment with the qubits (a phenomenon called *decoherence*[Chu95a] which leads to a loss of the stored quantum information) and the non-optimal control of the devices used during computation. It was shown, that these errors could be compensated through quantum error correction codes[Sho95,Ste96]. Further, it was also demonstrated that these errors could be compensated through so called fault-tolerant quantum computation schemes[Sho96] even if the devices used in the quantum computational system are themselves object of errors, provided that each quantum gate (i.e. the quantum equivalent of the logic gates used in classical electronics) has an error level below a certain threshold. As reviewed, for example, in [Nie03], many different bounds for this threshold have been demonstrated which differ on the assumptions made in the derivation, with typically reported values ranging around 10^{-4} to 10^{-6}.

In parallel to these studies, the other question which was persistently addressed after Shor’s discovery was which physical systems would have been suitable for implementing a quantum computer and a large number of proposals have since then been presented. Some of the first systems to be proposed were systems in which the information is memorized and processed through the interactions of photons[Chu95b], quantum computers exploiting cavity quantum electrodynamics[Tur95] or ion-traps [Cir95] and quantum computers based on nuclear magnetic resonance (NMR) [Cor97,Ger97]. Soon, these proposals were followed by the first experimental implementations of small quantum computational systems such as, for example, a two-qubit experiment based on the ion-trap proposal[Mon95] and two two-qubit experiments based on NMR[Chu98,Jon98]. However, despite the efforts, it became clear that even connecting just more than a few qubits, not to mention building a large-scale quantum computer able to factorise large integers, posed exceptional technological challenges which have not been solved yet [Nie03, Zol05]. To date, one of the most important demonstrations of quantum computational systems is probably still represented by the 7-qubit factorization algorithm implemented on an NMR quantum computer which factorized the number 15[Van01]. Considering that today's encryption keys are based on integers with more than 600 digits[Ger05] (corresponding to 2048 bits necessary for its digital codification), it is possible to see that quantum computation is still far away from implementing a computational system able to pose a threat to cryptographic systems. There are other applications, however, which would benefit from a quantum computer of smaller dimensions. A quantum computer of around 50 qubits would be able to simulate quantum mechanical systems intractable on a classical computer due to the exponential amount of information which needs to be stored and processed when describing such systems with classical bits[Spi06,Nie03]. Nevertheless, although some quantum computational systems larger than the 7-qubits experiment have been demonstrated (e.g. 12 qubits on an NMR quantum computer[Neg06]), scalable quantum computation has not been achieved yet, as will be discussed in more detail in Chapter 3, and it is not clear which physical implementation might prove best[Zol05].

One implementation technology which is believed to have the potential for high scalability is that of solid-state quantum computers in which qubits are stored in some form of spin system embedded in a semiconductor substrate since these proposals may benefit from the expertise and knowledge acquired through classical microelectronics[Cer05,Das05,Hog03]. These ideas were launched almost simultaneously through two different proposals, the one of qubits implemented in the spin of electrons trapped in quantum dots[Los98] and the one of qubits stored in the nuclear spin of phosphorus atoms embedded in a silicon substrate[Kan98], and have since then attracted much interest in the quantum computation research community, leading to a number of further proposals being presented (see [Vrj00,Wra01, Lad02], for example). The potential of some of these proposals has been also confirmed by measurements on their tolerance towards decoherence which have shown promising results[Bal09,Mor08, Tyr03]. However, to date, experimental implementations of these proposals have not managed to demonstrate systems larger than a few qubits[Pet05,Jel04] to the best of our knowledge. The difficulties in implementing larger systems are of different types and will be discussed in more detail in Chapter 3. In many cases, although single qubits may show promising features, one of the difficult problems to solve is how to introduce the interactions between qubits which are necessary for implementing a given quantum algorithm. In the pioneering proposal by Kane based on phosphorus atoms embedded in silicon[Kan98], for example, one of the problems seems to lie in the high-precision fabrication techniques required for the placement of control electrodes used to mediate the interactions between the qubits[Das05]. Similarly, qubits embedded in the electron spin of nitrogen-vacancies in diamond[Wra01] have demonstrated exceptional tolerance towards decoherence even at room temperature[Bal09] (whereas other implementations typically require operation temperatures below 10K[Pet05,Kan98,Tyr03]), indicating the potential for high-temperature quantum information processing. However, a scalable interaction mechanism between the qubits stored in the nitrogen-vacancy is not straightforward to implement.

In this context, a new model of quantum computer has been recently proposed which is based on the optically controlled, solid-state quantum gates proposed by Stoneham, Fisher and Greenland in [Sto03]. Throughout this thesis, following the terminology used in [Ker07], these gates will be referred to as *SFG gates* or, more generally, as to the *SFG model*. In the proposal, qubits are stored in the electron spin of a donor in a solid-state, possibly silicon, substrate. Interactions between two qubits are mediated through a control particle placed in their proximity. The distances between the particles are such that their wavefunctions are sufficiently separated to have negligible interaction. If, however, the control particle is excited by an optical pulse, its wavefunction will spread and overlap with those of the qubits leading to an effective interaction between them. The interaction is stopped when a second, de-exciting pulse is transmitted on the quantum register, bringing the control particle back to its ground state. In the ensemble of pulse interleave times T between exciting and de-exciting pulses, only a discrete subset generates entangling gates which leave the control particle unentangled from the qubits, avoiding loss of quantum information from the qubits to the controls. The potential of this implementation lies in the exploitation of the promising tolerance towards decoherence demonstrated by donor electron spin qubits[Tyr03] in conjunction with their optically mediated interaction mechanism obtained through the control defect. This interaction mechanism allows one to avoid the requirement of high-precision fabrication to ensure the exact placement of control electrodes necessary, for example, in Kane’s proposal[Kan98]. Also, this model may be compatible with the above mentioned quantum computation proposal based on diamond, using a control defect as the interaction link between the qubits implemented in the nitrogen-vacancy and could therefore operate above cryogenic temperatures[Sto09]. While it is expected that patches of up to 20 qubits may be controlled in a system based on SFG gates, larger quantum computers may then be built by the interconnection of different patches[Sto08].

The SFG model has been studied intensively, both theoretically and experimentally, in a project entitled “Putting the quantum into information technology” supported through the EPSRC Basic Technology program. This project was carried out at University College London, in collaboration between the Department of Physics and Astronomy and the Department of Electronic and Electrical Engineering, and the London Centre for Nanotechnology. As a result of the work conducted within the project, the details on the dynamics of the SFG quantum gate were presented in [Rod04], which also compared how accurately this model is able to produce two-qubit gates typically used in literature, such as, for example, the controlled-phase (or controlled-Z) gate[Nie03]. In [Ker07], these studies were further developed with the aim of identifying gate parameters able to produce both high-accuracy and fast two-qubit entangling gates. Experimentally, measurements on the life-times of excited states of phosphorus atoms in silicon have been recently performed which yielded important information on their behaviour as control particles[Vin08]. These results represent essential steps towards the implementation of an SFG quantum computer and lead the way to an experimental demonstration of a complete few-qubit quantum computation prototype, a fundamental test-bed for assessing the potential of this proposal.

The research work presented in this thesis was part of this project and focused on the problem of identifying quantum logic circuits suitable for the experimental demonstration of a small-scale quantum computational system based on the SFG model. Specifically, the overall aim of the research was to design quantum logic circuits assuming that a few-qubit quantum computational system had been realised and required verification.

To achieve this aim, the first step required the understanding of the main challenges of experimental quantum computation, as well as the identification of the algorithms used for testing prototypes of quantum computers and the typical system sizes, such as the number of qubits or the length and complexity of quantum circuits for the systems demonstrated to date. This study is described in Chapter 3. The analysis focused on some of the most studied implementations such as those based on NMR[Cor97,Ger97], ion-traps[Cir95] and Josephson junctions[Shn97], as well as proposals based on spin qubits in solid-state systems: quantum dots[Los98], donor spin qubits as in the Kane proposal[Kan98], the nitrogen-vacancy in diamond[Wra01] and silicon NMR[Lad02]. The review of the experiments conducted on these systems, identified the Deutsch-Jozsa algorithm[Deu92], and its refined version presented by Collins et al.[Col98], as a convenient mathematical problem often used for the experimental demonstration of few-qubit quantum computers (see [Chu98,Jon98,Kim00a,Fah08], for example). Even when performed on small quantum registers, this algorithm allows one to implement parallelism, interference and entanglement, the latter only if the algorithm, in its refined version, is performed on a quantum register of at least three qubits[Col98]. Aiming at identifying a test-problem which would prove that the analysed system effectively operates as a quantum computer while considering system parameters which would not lead, potentially, to insurmountable technological challenges, the three-qubit version of the refined Deutsch-Jozsa algorithm was chosen as the algorithm to develop in the research project here presented.

As previously mentioned, in [Rod04] it had been shown that quantum gates defined in the literature can be obtained through the SFG model and specific examples of SFG gates approximating quantum gates such as the controlled-phase gate had been reported. However, at the start of this work, a more general analysis of the ensemble of entangling gates which could be implemented within this proposal had not been performed yet. More specifically, with a view of designing quantum circuits based on SFG gates, it was not clear which alternatives were offered by this model in the choice of the gates to be used for the design of circuits. Whether, for example, it would have been more convenient to use the SFG model for obtaining gates which approximate the ones typically used in literature as shown in [Rod04] or whether there were other entangling SFG gates produced within this proposal which would have been worth considering. Hence, the next step of the research required to analyse the SFG model and to study the entangling characteristics of SFG gates, in order to identify which quantum gates would be available for the design of circuits based on this scheme.

After that, the issue of the design of quantum logic circuits was addressed. Similarly to classical circuit design, developing a quantum logic circuit requires one to find a sequence of quantum logic gates which implements the quantum computation one wishes to perform. These sequences typically comprise quantum gates which introduce interactions between qubits and operations on single qubits[DiV00]. Further, they are not unique and also depend on the type of quantum gates which the technology, or the physical system used to implement the quantum computer, can produce. Finally, in order to reduce the chances of errors accumulating along the computational path, the quantum circuits should be as short as possible[Bec96]. However, to develop quantum logic circuits it is also necessary to have convenient tools to analyse their performance. Specifically, one needs to be able to assess how well a given circuit implements the function it has been designed for. Also, even if a circuit implements a given function correctly, it is important to estimate how the circuit performance may change under non-ideal conditions such as, for example, fluctuations on the parameters relevant in the gate dynamics, which may manifest themselves in an experimental scenario. Algorithms for implementing numerical tools which simulate the behaviour of quantum logic circuits have been described, for example, in [Obe99,Sch00]. Given the input state of a computation register of qubits, these algorithms study the performance of a quantum logic circuit by analysing the impact of the quantum gates comprised in the circuit on the state of the qubits. However, in the SFG model, the interactions between qubits are mediated by control particles. As will be described in more detail in Chapters

4 and 5, the quantum gates implemented within this proposal also depend on the state of the control particle and its evolution must be considered in order to obtain a more complete picture when analysing the performance of circuits based on SFG gates. Hence, starting from the models presented, for example, in [Obe99,Sch00], it became clear that these protocols had to be further developed into a quantum logic simulator which would, firstly, follow the evolution of the control particles and, secondly, be based on the specific two-qubit quantum gates characteristic of the SFG proposal. This task required to identify how to incorporate the states of the control particles in the data to be processed by the simulator in order to have a numerical tool specifically tailored for circuits based on the SFG model and which could, therefore, be used for the study of how the SFG gate dynamics influences the result of the computation. The main characteristics of this quantum logic simulator are described in the first part of Chapter 5.

Further, what was also unknown at the start of the project was which techniques may have been used for deriving quantum circuits for a three-qubit refined Deutsch-Jozsa algorithm based on SFG gates. Quantum circuits implementing this algorithm for an NMR system had been reported, for example, in [Kim00a]. Hence, one possibility for obtaining circuits compatible with the SFG model would have been to adapt the circuits presented in [Kim00a] to SFG computation by approximating them through SFG gates. Nevertheless, it would also have been important to verify whether circuits with improved performance (e.g. circuits implementing the desired function with a higher precision or circuits characterised by a shorter computational time) could be obtained when deriving them directly for SFG gates, rather than adapting existing solutions developed for other implementations, and convenient methods for addressing this problem needed to be identified. In [Kim00a], for example, the circuits had been derived using the generator expansion technique[Kim00b] which addresses the design problem from the Hamiltonian describing the qubit interactions and the computation one wants to implement. However, because of the presence of the control particle in the description of the SFG dynamics, these methods cannot straightforwardly be applied to the SFG case. Also, a method like the generator expansion technique requires the quantum gate termed controlled-not (C-NOT)[Bar95] or the controlled-phase gate to be always part of the quantum gates library used during the design procedure. While these gates are widely used in literature, they are not the only entangling gates available.

Hence, in order to study how circuit topologies may change depending on the choice of the two-qubit entangling gates used during the design process, a technique which does not put any constraints on the quantum gate library was required. One technique which has this flexibility is the automated quantum circuit design process based on a genetic programming approach which was proposed by Williams and Gray[Wil99]. This technique has been successfully exploited for finding quantum circuits for one- and two-qubit Deutsch-Jozsa algorithms[Spe04,Sta06]. However, these circuits were oracle-based, i.e. the core of the algorithm was treated as a black box. Instead, when considering using the refined Deutsch-Jozsa algorithm for the experimental demonstration of a physical quantum computer, its complete decomposition into a sequence of gates comprising the one- and two-qubit operations realizable by the chosen technology is necessary. Hence, aiming at designing quantum circuits specifically tailored for SFG gates while testing how the choice of the gates used may influence the final topology of the resulting circuits, it became clear that these problems could have been addressed by implementing the automated quantum circuit design algorithm proposed by Williams and Gray for SFG computation, which led to the development of a second numerical tool described in the second part of Chapter 5.

The last part of the work described in this thesis addressed they key goal of the design of quantum circuits implementing a three-qubit refined version of the Deutsch-Jozsa algorithm[Col98] and suitable for the experimental demonstration of an SFG quantum computation prototype. The results of this study are presented in Chapter 6. Considering the exceptional technological challenges inherent in such an experimental demonstration, the aim was set on identifying strategies for obtaining the shortest possible circuits in order to minimise the possibilities of errors accumulating during computation and a number of different options for solving this task were identified. As mentioned above, one possibility would have been to adapt the circuits proposed in [Kim00a] to SFG computation. However, other solutions could be obtained by deriving circuits directly for SFG gates using the automated quantum circuit approach proposed by Williams and Gray. Nevertheless, for this strategy, it was not clear which SFG gates, out of the ensemble of gates which can be produced within this proposal, would have lead to convenient quantum circuits. Specifically, it was unknown whether it would have been best to use SFG gates approximating, for example, controlled-phase gates as shown in [Rod04] or whether improved circuits could have been derived using arbitrary entangling gates (meaning gates which do not have to resemble the ones typically used in literature such as, for example, the controlled-phase gate or the C-NOT gate) characterised by a shorter gate computation time compared to approximations of controlled-phase gates. Also, the choice of using the SFG model for approximating controlled-phase gates still leaves a number of options open regarding the precision of the approximation or the gate computation time of the corresponding SFG gates. Hence, it became clear that different design strategies (different also in the SFG gates used in the design procedure) had to be tested and compared. Further, considering that these circuits were derived in the perspective of an experimental implementation, the next step of the research addressed the problem of analysing the performance of the best performing solution under increasing fluctuations of the parameters relevant in the SFG gate dynamics.

Finally, in an SFG-based quantum computer, the two-qubit interactions would be triggered through sequences of optical pulses. Examples of recently proposed control particle systems[Sto08] are the double donor Se^+ in silicon and phosphorus particles in diamond which would both require excitation wavelengths around 2.2-2.3μm [Ber89,Laz08]. Moreover, as described in [Rod04], pulsewidths necessary for correctly implementing the SFG protocol are of the order of a picosecond. Thus, to implement quantum circuits based on the SFG model for the two control particle systems mentioned above would require the generation of a sequence of picosecond pulses in a wavelength range around 2.2-2.3μm. At the start of the work, convenient optical systems for implementing such a signal had not been analysed yet and the final part of the work on the quantum circuit design problem was therefore dedicated to exploring systems able to produce the control signal necessary for the implementation of the presented circuits.

1.1 Outline of the thesis

The rest of the thesis is arranged as follows: Chapter 2 gives a brief introduction to some aspects of quantum computation and quantum information processing while Chapter 3 reviews physical implementations of quantum computers. In Chapter 4, the SFG gate is described and its entangling characteristics analysed. Chapter 5 describes the two numerical tools which have been developed throughout the project. The developed quantum circuits are presented in Chapter 6 as well as the optical systems which may implement them through the appropriate optical pulse sequences. Finally, the main results of this research as well as possibilities for future developments are summarised in Chapter 7.

1.2 Original contributions of the thesis

The following original contributions were made in the course of the research here presented:

- Development of a quantum logic simulator, specifically designed for the SFG model, which, a part from modelling the evolution of qubits during a quantum computation, also takes into account the state of the control particles.
- Visualisation of the entangling characteristic of SFG gates[1] and study of the quantum gates which can be produced within the SFG model. This analysis laid the foundations for the choice of the gates subsequently used in [2]-[4].
- Development of quantum logic circuits implementing a three-qubit refined Deutsch-Jozsa algorithm specifically tailored for a quantum computation system based on the SFG model[2]-[4].
- Assessment of the tolerance towards fluctuations in the SFG gate parameters of one of the circuits solving the refined Deutsch-Jozsa algorithm[3].
- Initial design for an optical system able to produce the control signal necessary for implementing the proposed quantum circuits[3].

1.3 Publications and conference presentations arising from the work presented in this thesis

[1] A.Del Duce, S.Savory, P.Bayvel: "Design and optimisation of quantum logic circuits for a three-qubit Deutsch-Jozsa algorithm implemented with optically-controlled, solid-state quantum logic gates", arXiv:0910.1673v1 [quant-ph]

[2] A. Del Duce, S.Savory, P.Bayvel: "Implementation of a three-qubit refined Deutsch-Jozsa algorithm using SFG quantum logic gates", *Journal of Physics: Condensed Matter*, vol.18, S795-S805, (2006)

[3] A. Del Duce, P.Bayvel: "Quantum logic circuits and optical signal generation for a three-qubit, optically-controlled, solid-state quantum

computer", accepted for publication in the *IEEE Journal of Selected Topics in Quantum Electronics*, issue on Quantum Communications and Information Science.

[4] A. Del Duce, P.Bayvel: "Design of quantum logic circuits for a three-qubit refined Deutsch-Jozsa algorithm with optically controlled, solid-state quantum logic gates", *Proceedings of the London Communication Symposium 2009*: http://www.ee.ucl.ac.uk/lcs/previous/LCS2009/LCS/lcs09_17.pdf, (2009)

[5] A.Del Duce, P.Bayvel: "Design of quantum logic circuits and optical signal generation for a refined Deutsch-Jozsa algorithm with optically controlled, solid-state quantum gates", presentation at the IoP *Young Researchers in Optics Meeting*, Imperial College, London, 16th September, (2009)

1.4 References

[Bal09] G.Balasubramanian, P.Neumann, D.Twitchen, M.Markham et al.: "Ultralong spin coherence time in isotopically engineered diamond", *Nature Materials*, vol.8, pp. 383-387, (2009)

[Bar95] A. Barenco, D.Deutsch, A.Ekert, R.Jozsa: "Conditional quantum dynamics and logic gates", *Physical Review Letters,* vol.74, pp.4083-4086,(1995)

[Bec96] D.Beckman, A.N.Chari, S.Devabhaktuni, J.Preskill, "Efficient networks for quantum factoring", *Physical Review A*, vol.54, pp. 1034-1063, (1996)

[Ber89] K. Bergman, G. Grossmann, H.G. Grimmeiss, M. Stavole, R.E. McMurray, "Applicability of the deformation-potential approximation to deep donors in silicon", *Physical Review B*, vol. 39, pp.1104-1119, (1989)

[Cer05] V.Cerletti, W.A.Coish et al.:"Recipes for spin-based quantum computing", *Nanotechnology*, vol.16, R27-R49, (2005)

[Chu95a] I.L.Chuang, R.Laflamme, P.W.Shor, W.H.Zurek: "Quantum computers, factoring, and decoherence", *Science*, vol.270, pp.1633-1635, (1995)

[Chu95b] I.L.Chuang, Y.Yamamoto: "Simple quantum computer", *Physical Review A*, vol.52, pp.3489-3496, (1995)

[Chu98] I.L. Chuang, L.M.K. Vandersypen et al. : "Experimental realization of a quantum algorithm", *Nature*, vol. 393, pp.143-146, (1998)

[Cir95] J.I.Cirac, P.Zoller: "Quantum computation with cold trapped ions", *Physical Review Letters*, vol.74,(20),pp.4091-4094,(1995)

[Col98] D. Collins, K.W. Kim, W.C. Holton, "Deutsch-Jozsa algorithm as a test for quantum computation", *Physical Review A,* vol. 58, Sept.1998, pp. R1633-R1636

[Cor97] D. G. Cory, A. F. Fahmy et al.: "Ensemble quantum computing by NMR spectroscopy", *Proceedings of the National Academy of Sciences of the United States of America,* vol.94, pp.1634-1639, (1997)

[Das05] S.Das Sarma, R.de Sousa et al.:"Spin quantum computation in silicon nanostructures", *Solid State Communication*, vol.133, pp.737-746, (2005) [Deu85] D.Deutsch: “Quantum Theory, the Church-Turing Principle and the Universal Quantum Computer”, *Proceedings of the Royal Society of London, Series A, Mathematical Physical and Engineering Sciences*, vol.400, pp.97-117, (1985)

[Deu92] D.Deutsch, R.Jozsa, “Rapid solution of problems by quantum computation”, *Proceedings of the Royal Society of London Series A-Mathematical Physical and Engineering Sciences*, vol.439, pp.553-558, (1992)

[DiV00] D.P.DiVincenzo: “The physical implementation of quantum computation”, *Fortschritte der Physik*, vol.48, pp.771-783, (2000)

[Fah08] A.F.Fahmy, R.Marx, W.Bermel, S.J.Glaser, “Thermal equilibrium as an initial state for quantum computation by NMR”, *Physical Review A*, vol.78, Article number 022317, (2008)

[Fey82] R.P.Feynman: “Simulating physics with computers”, *International Journal of Theoretical Physics*, vol.21, pp.467-488, (1982)

[Ger97] N. Gershenfeld, I. L. Chuang: “Bulk spin-resonance quantum computation”, *Science*, vol.275, pp.350-356, (1997)

[Ger05] E.Gerjuoy: “Shor’s factoring algorithm and modern cryptography. An illustration of the capabilities inherent in quantum computers”, *American Journal of Physics*, vol.73, pp.521-540, (2005)

[Gro97] L.K.Grover; “Quantum mechanics helps in searching for a needle in a haysack”, *Physical Review Letters*, vol.79, pp.325-328, (1997)

[Hog03] J. Hogan, “Computing – Quantum bits and silicon chips”, *Nature*, vol.424, pp.484-486, (2003)

[Jel04] F.Jelezko, T.Gaebel et al.: “Observation of coherent oscillation of a single nuclear spin and realization of a two-qubit conditional quantum gate”, *Physical Review Letters*, vol.93, Article number 130501, (2004)

[Jon98] J. A. Jones, M. Mosca: “Implementation of a quantum algorithm on a nuclear magnetic resonance quantum computer”, *Journal of Chemical Physics*, vol.109, pp.1648-1653, (1998)

[Kan98] B.E. Kane: “A silicon-based nuclear spin quantum computer”, *Nature*, vol.393, pp.133-137,(1998) [Ker07] A. Kerridge, A.H. Harker, A.M. Stoneham, “Electron dynamics in quantum gate operation”,*Journal of Physics: Condensed Matter*, vol. 19, Article number 282201, (2007)

[Kim00a] J. Kim, J.S. Lee, S. Lee, C. Cheong, “Implementation of the refined Deutsch-Jozsa algorithm on a three-bit NMR quantum computer”, *Physical Review A*, vol. 62, Article number 022312, (2000)

[Kim00b] J.Kim, J.S.Lee, S.Lee: “Implementing unitary operators in quantum computation”, *Physical Review A*, vol.61, Article number 032312, (2000)

[Lad02] T.D.Ladd, J.R.Goldman et al.: “All-silicon quantum computer”, *Physical Review Letters*, vol.89, Article number 017901, (2002)

[Laz08] A.Lazea, V.Mortet, J.D’Haen, P. Geithner, J. Ristein, M.D’Olieslaeger, K. Haenen, “Growth of polycrystalline phosphorus-doped CVD diamond layers”, *Chemical Physics Letters*, vol. 454, pp.310-313, (2008)

[Llo96] S.Lloyd: “Universal quantum simulators”, *Science*, vol.273, pp.1073-1078, (1996)

[Los98] D.Loss, D.P.DiVicenzo: “Quantum computation with quantum dots”, *Physical Review A*, vol.57, p.120-126, (1998)

[Mon95] C.Monroe, D.M.Meekhof et al.:“Demonstration of a fundamental quantum logic gate”, *Physical Review Letters*, vol.75, pp.4714-4718, (1995)

[Mor08] J.J.L.Morton, A.M.Tyryshkin, R.M.Brown, S.Shankar, B.W.Lovett et al.: “Solid-state quantum memory using the ^{31}P nuclear spin”, *Nature*, vol.445, pp.1085-1088, (2008)

[Neg06] C.Negrevergne et al.: “Benchmarking quantum control methods on a 12-qubit system”, *Physical Review Letters*, vol.96, Article number 170501, (2006)

[Nie03] M.A. Nielsen, I.L.Chunag: “Quantum computation and quantum information”, Cambridge: Cambridge University Press, (2003)

[Obe99] K.M. Obenland, A.M.Despain, “Simulating the effect of decoherence and inaccuracies on a quantum computer”, *in Quantum Computing and Quantum Communications, Book Series: Lecture Notes in Computer Science*, vol. 1509, pp.447-459, (1999) [Pet05] J.R.Petta, A.C.Johnson et al.: “Coherent manipulations of coupled electron spins in semiconductor quantum dots”, *Science*, vol.309, p.2180-2184, (2005)

[Rod04] R. Rodriquez, A.J. Fisher, P.T.Greenland, A.M. Stoneham, “Avoiding entanglement loss when two-qubit quantum gates are controlled by electronic excitation”, *Journal of Physics: Condensed Matter*, vol.16, pp.2757-2772, (2004)

[Sch00] F. Schürmann: “Interactive Quantum Computation”, Master Thesis, submitted in September 2000, University of New York at Buffalo.

[Sch95] B.Schumacher, “Quantum coding”, *Physical Review A*, vol.51, pp.2738-2747, (1995)

[Shn97] A.Shnirman, G. Schön et al.:“Quantum manipulations of small Josephson junctions”, *Physical Review Letters*, vol.79, pp.2371-2374,(1997)

[Sho94] P.W. Shor: “Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer”, *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*, edited by S.Goldwasser, (IEEE Computer Society, Los Alamos, Ca), pp.124-134, (1994)

[Sho95] P.W.Shor, “Scheme for reducing decoherence in quantum computer memory”, *Physical Review A*, vol.52, Oct. 1995, pp.R2493-2496

[Sho96] P.W.Shor, “Fault-tolerant quantum computation”, *Proceeding of the 37th Annual Symposium on Foundations of Computer Science*, Book Series: Annual Symposium on Foundations of Computer Science, pp.56-65, (1996)

[Spe04] L. Spector, “Automatic Quantum Computer Programming: A Genetic Programming Approach”, Boston: Kluwer Academic Publishers, 2004

[Spi06] T.P.Spiller, W.J.Munro: “Towards a quantum information technology industry”, *Journal of Physics: Condensed Matter*, vol.118, pp.V1-V10, (2006)

[Sta06] R. Stadelhofer, “Evolving Blackbox Quantum Algorithms using Genetic Programming”, Dissertation submitted to the Department of Physics, University of Dortmund, Germany, 2006

[Ste96] A.M.Steane, “Error correcting codes in quantum theory”, *Physical Review Letters*, vol.77, Jul. 1996, pp.793-797

[Sto03] A.M.Stoneham, A.J.Fisher, P.T.Greenland: “Optically driven silicon-based quantum gates with potential for high-temperature operation”, *Journal of Physics: Condensed Matter*, vol.15, L447-L451, (2003)

[Sto08] A.M. Stoneham, “The quantum in your materials world”, *Materials Today*, vol.11, pp. 32-36, (2008)

[Sto09] A.M. Stoneham, A.H.Harker, G.W.Morely: “Could one make a diamond-based quantum computer”, *Journal of Physics: Condensed Matter*, vol.21, Article number 364222,(2009)

[Tur95] Q.A.Turchette, C.J.Hood et al.: “Measurement of conditional phase shifts for quantum logic”, *Physical Review Letters*, vol.75, pp.4710-4713, (1995)

[Tyr03] A.M. Tyryshkin, S.A. Lyon, A.V.Astashkin, A.M. Raitsimring: “Electron spin relaxation times of phosphorus donors in silicon”, *Physical Review B*, vol. 68, Article number 193207, (2003)

[Van01] L.M.K.Vandersypen, M.Steffen et al.: “Experimental realization of Shor’s quantum factoring algorithm using nuclear magnetic resonance”, *Nature*, vol.414, pp. 883-887, (2001)

[Vin08] N.Q. Vinh, P.T.Greenland, K. Litvinenko, B. Redlich, A.F.G van der Meer, S.A. Lynch et al.: “Silicon as a model ion trap: Time domain measurements of donor Rydberg states”, *Proceedings of the National Academy of Sciences of the United States of America*, vol.105, pp.10649-10653, (2008)

[Vri00] R.Vrijen, E.Yablonovitch et al.: “Electron-spin-resonance transistors for quantum computing in silicon-germanium heterostructures”, *Physical Review A*, vol.62, Article number 012306, (2000)

[Wil99] C.P.Williams, A.G.Gray: “Automated design of quantum circuits”, *in Quantum Computing and Quantum Communications, Book Series: Lecture Notes in Computer Science*, vol.1509, pp.113-125,(1999)

[Wra01] J.Wrachtrup, S.Ya.Kilin, A.P.Nizvtsev, “Quantum computation using the 13C nuclear spins near the single NV defect center in diamond”, *Optics and Spectroscopy*, vol.91, pp.429-437, (2001)

[Zol05] P.Zoller, T.Beth et al.: “Quantum information processing and communication - Strategic report on current status, visions and goals for research in Europe”, *European Physical Journal D*, vol.36, pp.203-228, (2005)

Chapter 2 Theoretical and experimental aspects of quantum computation

This chapter introduces the basic theory of quantum information processing on which the remainder of the thesis is built. First, the main characteristics of quantum computation will be described. Then, some experimental aspects of quantum computation are introduced in Section 2.2. Finally, the concept of quantum logic circuit design is discussed in Section 2.3.

2.1 Introduction to quantum computation

The elementary unit of classical digital information processing is the *bit*[Sha48], the binary unit of information, which can be either in the state 0 or 1. Considering a register of N classical bits, in which each bit b_i can be either in the state 0 or 1 and in which the first bit is associated to the value $b_0 \cdot 2^0$, the second one to $b_1 \cdot 2^1$ and so on up to $b_{N-1} \cdot 2^{N-1}$, this system can be used to store or process one out of 2^N values:

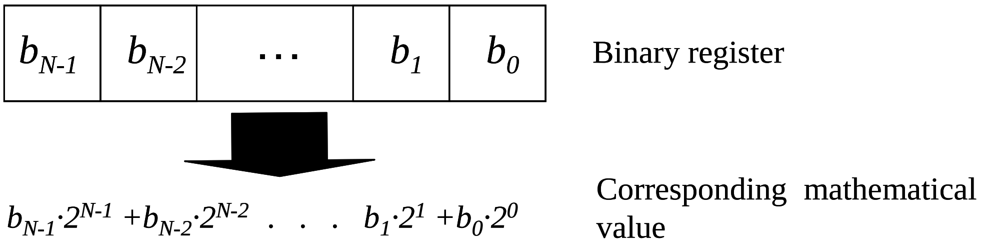

b_{N-1}
b_{N-2}
...
b_1
b_0
Binary register
$b_{N-1} \cdot 2^{N-1} + b_{N-2} \cdot 2^{N-2}$ . . . $b_1 \cdot 2^1 + b_0 \cdot 2^0$
Corresponding mathematical value

Figure 2-1:Classical binary information register

The term classical computer will be used when referring to a computational system based on the information register described above. At any time in such a system we can measure the state of any bit b_i of the register, invert its value by applying a NOT gate which transforms the value 1 in 0 and vice versa, or apply mathematical functions between bits through logical gates such as the NAND gate (see, for example, [Nie03]). A characteristic of classical computation and information processing is that most of the systems are organised on a sequential base. This means that, given a function $f_N(x)$ which has to be applied to a group of input elements x_i, the same function must be applied to each element of the input data: The first element of the input data set is stored into a computation register, the function $f_N(x)$ is applied through a sequence of logic gates, the final result is stored in a further output register and the same procedure is repeated to each element of the input data set. Finally, the output data can be compared to extrapolate the desired information.

In quantum information processing the elementary unit of information is a two-level quantum mechanical system called quantum bit or qubit, a term first introduced in [Sch95]. The information ‘0’ is associated with one of the two states, which is then labelled $|0\rangle$, while the other one is labelled $|1\rangle$. It will be discussed later how qubits can be implemented in physical systems. In the meantime it is important to note that, independently of the physical system which implements it, a qubit is a two-level quantum mechanical system and it can, therefore, not only exist in either the state $|0\rangle$ or $|1\rangle$, but also in a superposition

$$\left|\psi_q\right\rangle = \alpha|0\rangle + \beta|1\rangle \tag{2.1}$$

of its two states, where α and β are two complex coefficients which satisfy[Deu85]:

$$|\alpha|^2 + |\beta|^2 = 1 \tag{2.2}$$

Firstly, equation (2.1) shows that a qubit can simultaneously store the information 0 and 1. Also, from equation (2.2) it can be seen that these coefficients define a continuum of states in which the qubit can be between the states $|0\rangle$ and $|1\rangle$. To visualise the space defined by a qubit it is useful to use the Bloch Sphere representation[Nie03]. Using equation (2.2), equation (2.1) can be rewritten as:

$$\left|\psi_q\right\rangle = e^{i\gamma}\left(\cos\left(\frac{\vartheta}{2}\right)|0\rangle + e^{i\varphi}\sin\left(\frac{\vartheta}{2}\right)|1\rangle\right) \tag{2.3}$$

Neglecting the common phase factor which does not carry any information on the relative position of the state $|0\rangle$ with respect to the state $|1\rangle$, equation (2.3) allows one to describe the state of a qubit through the two angles ϑ and φ which describe the three dimensional sphere with a unit radius, termed the Bloch sphere, shown in Figure 2-2.

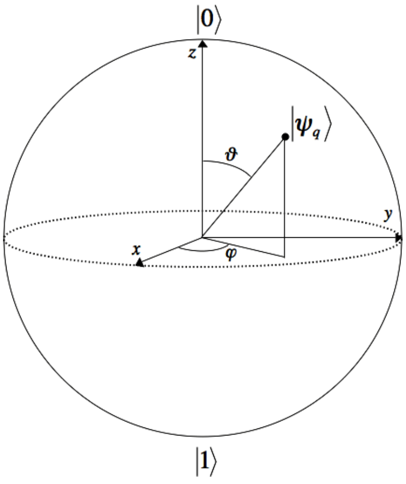


Figure 2-2: Qubit representation in the Bloch Sphere

It can be seen that a qubit may occupy any point on the surface of the sphere. Hence, as will be seen later, to change the state of a qubit corresponds to a rotation of its state on the Bloch sphere.

In terms of notation, it can also be seen that the state of the qubit is completely defined by the vector:

$$\begin{bmatrix}\alpha\\ \beta\end{bmatrix} \tag{2.4}$$

Hence, a qubit defines a complex vector space of two dimensions.

Considering now two qubits, each in an equal superposition, i.e. with $\alpha = \beta = 1/\surd 2$, of $|0\rangle$ and $|1\rangle$, their combined state can be described as:

$$\begin{aligned}&|\psi_{q1}\rangle|\psi_{q0}\rangle = \left(\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle\right)\left(\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle\right) = \\ &= \frac{1}{2}\big(|00\rangle + |01\rangle + |10\rangle + |11\rangle\big)\end{aligned} \tag{2.5}$$

Using the same notation used in Figure 2-1 for each of the four words $|qb_1qb_0\rangle$ in the last line of expression (2.5), one gets:

$$|\psi_{q1}\rangle|\psi_{q0}\rangle = |\psi_{q1}\psi_{q2}\rangle = \frac{1}{2}\big(|0\rangle + |1\rangle + |2\rangle + |3\rangle\big) \tag{2.6}$$

Equation (2.6) shows that two qubits in an equal superposition of the states $|0\rangle$ and $|1\rangle$ can be used to store simultaneously four values. Using the vector notation introduced in expression (2.4), a new vector can be created using the Kronecker product to describe the combined state expressed in (2.6)[Nie03]:

$$\begin{bmatrix}\alpha_1\\ \beta_1\end{bmatrix} \otimes \begin{bmatrix}\alpha_0\\ \beta_0\end{bmatrix} = \begin{bmatrix}\alpha_1\alpha_0\\ \alpha_1\beta_0\\ \beta_1\alpha_0\\ \beta_1\beta_0\end{bmatrix} = \begin{bmatrix}c_0\\ c_1\\ c_2\\ c_3\end{bmatrix} \tag{2.7}$$

The vector space defined by two qubits is then a four dimensional complex vector space.

Generalising the above presented discussion, a quantum register comprising *N* qubits, with each qubit in an equal superposition of $|0\rangle$ and $|1\rangle$, is described by the state:

$$\left|\psi_{q(N-1)}\psi_{q(N-2)}..\psi_{q1}\psi_{q0}\right\rangle = \frac{1}{\sqrt{2^N}}\Big(\,|11...11\rangle + |11...10\rangle + ... \\ + |00..01\rangle + |00..00\rangle \,\Big) \qquad (2.8)$$

Thanks to the superposition of each qubit, this register can be used to store simultaneously 2^N values. Even more, without worrying for the moment about how this can be practically done, if we apply to such a register a quantum function $qf_N(x)$, then this function is automatically applied to all values stored in the quantum register. There is no need to repeat any procedure 2^N times, as would be necessary in the classical case. Hence, quantum information processing naturally implements parallel computation.

Using again the Kronecker product for all N qubits:

$$\begin{bmatrix}\alpha_{N-1}\\ \beta_{N-1}\end{bmatrix} \otimes \begin{bmatrix}\alpha_{N-2}\\ \beta_{N-2}\end{bmatrix} \otimes ... \otimes \begin{bmatrix}\alpha_{1}\\ \beta_{1}\end{bmatrix} \otimes \begin{bmatrix}\alpha_{0}\\ \beta_{0}\end{bmatrix} = \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{2^N-2} \\ c_{2^N-1}\end{bmatrix} \qquad (2.9)$$

it can be seen that such a register defines a complex vector space of dimension 2^N. The term quantum computer will be used to define a computational system in which the information is stored or processed using quantum registers as the one described above.

Although it is possible to store a vast amount of information in a quantum register, it cannot be accessed straightforwardly. As summarised in [Nie03], when measured, the superposition of a qubit collapses into one of the two states $|0\rangle$ or $|1\rangle$ with probability $|\alpha|^2$ and $|\beta|^2$, respectively. Considering a quantum register of N qubits which after a computation is in some superposition of its storable numbers, then the act of measurement of the register and the collapse of the superposition of each qubit will result in obtaining one out of all the values which were stored prior to the measurement. This means that a quantum computer is a probabilistic computer which typically exploits repetition of the computation to enhance the probability of success [Deu92].

In classical computation, one wants to apply a mathematical function to a set of input data stored in a register. The implementation of the function is achieved by decomposing it into a sequence of operations on the single bits and by applying logic gates between bits. Similar processes apply to quantum computation. To perform a quantum computation means to apply some mathematical function to a quantum register and this function is implemented through a sequence of quantum transformations on the single qubits and by controlling interactions between them typically through two-qubit quantum logic gates [DiV95,DiV00].

The dynamics of these transformations on the state $|\psi\rangle$ of a quantum register are governed by the Schrödinger equation[Nie03]:

$$i\hbar\frac{\partial|\psi(t)\rangle}{\partial t}=H_{Ham}|\psi(t)\rangle \tag{2.10}$$

where the Hamiltonian H_{Ham} contains the information on the action which is taken on the qubits and $\hbar$ is Planck's constant. For simplicity, $\hbar=1$ will be used in the reminder of the discussion. In many cases H_{Ham} can be described through a time-independent operator and equation (2.10) has solution:

$$|\psi(t_1)\rangle=e^{-iH_{Ham}(t_1-t_0)}|\psi(t_0)\rangle=U|\psi(t_0)\rangle;\quad \text{where } U=e^{-iH_{Ham}(t_1-t_0)} \tag{2.11}$$

Hence, given the initial state $|\psi(t_0)\rangle$ of a quantum register at a time interval t_0, using equation (2.11) it is possible to evaluate the state of the register $|\psi(t_1)\rangle$ at a time interval t_1 after an interaction corresponding to a specific quantum transformation. The operator U is a unitary operator, i.e. such that $U^{\dagger}U=I$, where '$\dagger$' indicates the complex conjugate of the transpose of a matrix and I is the identity matrix:

$$I=\begin{bmatrix}1 & 0 & \cdots & 0 & 0\\ 0 & 1 & 0 & \cdots & 0\\ \vdots & 0 & \ddots & \ddots & \vdots\\ \vdots & \vdots & \ddots & 1 & 0\\ 0 & 0 & \cdots & 0 & 1\end{bmatrix} \tag{2.12}$$

For the case of single- and two-qubit quantum logic gates, U operators are, respectively, described by 2x2 and 4x4 matrices.

A quantum computation is typically described by a well defined sequence of single- and two-qubit operations which, once applied to the quantum register initialized to the state $|\psi_{input}\rangle$, implement the desired algorithm [DiV00]:

$$|\psi_{output}\rangle=U_mU_{m-1}\ldots\ldots.U_2U_1|\psi_{input}\rangle \tag{2.13}$$

where $|\psi_{output}\rangle$ is the output state of the quantum register at the end of the computation and U_m is a unitary matrix describing either a single- or a two-qubit transformation.

As will be described later, these transformations correspond from the experimental point of view to physical actions applied to the quantum mechanical systems used to store the qubits in order to change their states.

2.1.1 Single-qubit operations

Single qubit operations are the equivalent of the NOT gate in classical electronics which transforms a bit of value “1” in “0” and vice versa. However, while the inversion of the value is the only possible action which can be taken on a bit, as this one can only be in the state 0 or 1, in quantum computation, because the qubit can exist in a continuum of states, a continuum of single qubit operations is available. This can be seen considering the single qubit operation *X*, a unitary operator described mathematically by:

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \tag{2.14}$$

When applying it to a single qubit starting in the state $|0\rangle$ one obtains:

$$|0\rangle \rightarrow \begin{bmatrix} \alpha = 1 \\ \beta = 0 \end{bmatrix}; \quad X\begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \rightarrow |1\rangle \tag{2.15}$$

The *X* gate performs a similar inversion to the one implemented by the NOT gate in the classical case. By looking at Figure 2-2, it can be seen that the *X* gate takes the $|0\rangle$-state, which is located on the north pole of the sphere, and transforms it into the $|1\rangle$-state which lies on the opposite pole. More generally, when used on an arbitrary state $\begin{bmatrix} \alpha \\ \beta \end{bmatrix}$, the *X* gate switches the two parameters giving the new state $\begin{bmatrix} \beta \\ \alpha \end{bmatrix}$. By using expression (2.3), and considering that sin(*x*)=cos(*π/2*-*x*) and cos(*x*)=sin(*π/2*-*x*), it can be seen that this transformation corresponds to:

$$X\begin{bmatrix}\cos\left(\frac{\vartheta}{2}\right)\\ e^{i\varphi}\sin\left(\frac{\vartheta}{2}\right)\end{bmatrix}=\begin{bmatrix}e^{i\varphi}\sin\left(\frac{\vartheta}{2}\right)\\ \cos\left(\frac{\vartheta}{2}\right)\end{bmatrix}=e^{i\varphi}\begin{bmatrix}\cos\left(\frac{\pi-\vartheta}{2}\right)\\ e^{-i\varphi}\sin\left(\frac{\pi-\vartheta}{2}\right)\end{bmatrix}$$
$$=e^{i\varphi}\begin{bmatrix}\cos\left(\frac{\vartheta'}{2}\right)\\ e^{i\varphi'}\sin\left(\frac{\vartheta'}{2}\right)\end{bmatrix}\rightarrow\begin{matrix}\vartheta'=\pi-\vartheta\\ \varphi'=-\varphi\end{matrix} \tag{2.16}$$

which is equivalent in the Bloch Sphere representation to rotating the qubit state around the *x*-axis of an angle π. Because single qubit operations rotate the state of a qubit on the Bloch Sphere, they are often referred to as single qubit rotations[Nie03]. As will be described in more detail in Section 2.2.1 the *X* rotation can be implemented on a qubit through an Hamiltonian of the form:

$$H_{Ham}=\frac{\omega_x}{2}X \tag{2.17}$$

which corresponds to the unitary transformation:

$$R_x(\vartheta)=e^{-i\frac{\omega_x t}{2}X}=e^{-i\frac{\vartheta}{2}X} \tag{2.18}$$

Without getting into the details of how this interaction can be initiated (which will be described in Section 2.2.1), for an interaction time such that $\omega_x t=\pi$, expression (2.18) returns $R_x(\pi)=X$. More generally, for a different interaction time *t*, any rotation around the *x*-axis in the Bloch sphere representation can be achieved. Moreover, changing the amount of rotation and the rotation axis, an infinite number of single qubit operations can be defined. A very important single qubit gate is the Hadamard gate *H*:

$$H=\frac{1}{\sqrt{2}}\begin{bmatrix}1&1\\1&-1\end{bmatrix} \tag{2.19}$$

When applied, to the states $|0\rangle$ or $|1\rangle$, this gate transforms the single state in an equal superposition:

$$\begin{aligned}H|0\langle&=\frac{1}{\sqrt{2}}\begin{bmatrix}1&1\\1&-1\end{bmatrix}\begin{bmatrix}1\\0\end{bmatrix}=\frac{1}{\sqrt{2}}\begin{bmatrix}1\\1\end{bmatrix}=\frac{1}{\sqrt{2}}(|0\langle+|1\langle)\\ H|1\langle&=\frac{1}{\sqrt{2}}\begin{bmatrix}1&1\\1&-1\end{bmatrix}\begin{bmatrix}0\\1\end{bmatrix}=\frac{1}{\sqrt{2}}\begin{bmatrix}1\\-1\end{bmatrix}=\frac{1}{\sqrt{2}}(|0\langle-|1\langle)\end{aligned} \tag{2.20}$$

Such gates are present at the beginning of most quantum algorithms and are used to bring all the qubits of the quantum register, which are typically initialised to the sate $|0\rangle$

or $|1\rangle$, to an equal superposition of their states in order to load the register with all possible 2^N values.

The Hadamard gate is also important, because it demonstrates the feature of interference in quantum computation. Considering a qubit in an equal superposition of its states $|0\rangle$ and $|1\rangle$, when applying a Hadamard gate one obtains:

$$\begin{aligned} &H\frac{1}{\sqrt{2}}\left(|0\rangle+|1\rangle\right)=\frac{1}{\sqrt{2}}\left(H|0\rangle+H|1\rangle\right)= \\ &=\frac{1}{\sqrt{2}}\left(\frac{1}{\sqrt{2}}\left(|0\rangle+|1\rangle\right)+\frac{1}{\sqrt{2}}\left(|0\rangle-|1\rangle\right)\right)=|0\rangle \end{aligned} \tag{2.21}$$

From expression (2.21) it can be seen how the $|1\rangle$-components with opposite sign cancel each other out and vanish. This interference phenomenon is another of the powerful features of quantum computation as it can be used during computation to filter out erroneous solutions from the computation register[Cle98].

2.1.2 Two-qubit gates and universality

To perform mathematical functions on a quantum register one needs not only to control the state of a single qubit, but also to implement interactions between qubits similar to the logic gates used in classical electronics[Deu89]. Moreover, in order to be able to implement an arbitrary quantum computation, one needs a *universal* set of gates, i.e. a set of gates able to produce any unitary transformation[Nie03]. An important two-qubit gate in quantum computation is the controlled-not gate (C-NOT). Given two qubits *q0* and *q1*, the C-NOT gate produces the transformation $|q1,q0\rangle \xrightarrow{C-NOT} |q1,q1\oplus q0\rangle$ (where $\oplus$ is addition modulo 2), which is the quantum generalization of the classical XOR gate[Nie03,DiV00]. In [Bar95], it was shown that the C-NOT gate together with single-qubit operations forms a universal set of gates and since then it is often used as a reference gate in the demonstration of experimental quantum computational systems or for the description of quantum logic circuits (see, for example, [Sch03,Nie03]). Nevertheless, as for the case of single-qubit operations, there are a number of other two-qubit gates and almost contemporarily to the demonstration of the universality of the C-NOT gate it was also shown that almost any two-qubit gate together with single-qubit operations forms a universal set of gates[Deu95,DiV95]. This result is extremely valuable since not all physical systems may directly implement a C-NOT gate but can nevertheless achieve the full power of quantum computation[Div00]. While other quantum gates are discussed in the reminder of this thesis, the C-NOT gate will now be described in more detail in order to familiarise with the main features of two-qubit quantum gates.

The unitary matrix describing the C-NOT gate is shown in expression (2.22):

$$C-NOT=\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 1\\ 0 & 0 & 1 & 0\end{bmatrix} \qquad (2.22)$$

From expression (2.22), it can be seen that the C-NOT gate takes following actions on the states of a two-qubit quantum register:

$$\begin{aligned}|00\rangle &\rightarrow |00\rangle\\ |01\rangle &\rightarrow |01\rangle\\ |10\rangle &\rightarrow |11\rangle\\ |11\rangle &\rightarrow |10\rangle\end{aligned} \qquad (2.23)$$

When labelling the states in expression (2.23) $|qb_1\, qb_0\rangle$, it is possible to see that the first two states remain untouched and are characterised by having qubit qb_1 in state 0. The last two states, however, are characterised by having qb_1 in state $|1\rangle$ and are subject to the inversion of the value stored by qb_0. Hence, in a C-NOT gate, the state of a target qubit (in this case qb_0) is inverted only if the control qubit (in this case qb_1) is in state $|1\rangle$.

Figure 2-3 shows a C-NOT gate in the quantum circuit representation typically used in quantum computation. The circuit is read from left to right and describes the sequence of gates (in this case only the C-NOT gate) which is applied to the quantum register q_0q_1. The evolution of the circuit can be analysed following the *wires* (horizontal lines) and the quantum gates encountered along them. Single-qubit operations are usually represented by a box on the wire with a label identifying the specific type of gate while in two-qubit interactions, as shown in Figure 2-3, a symbol connects the wires corresponding to the qubits which are made to interact by the quantum gate. In the case of the C-NOT gate, for example, a circular symbol indicates the target qubit of the quantum gate while a connection through a vertical wire describes the corresponding control qubit.

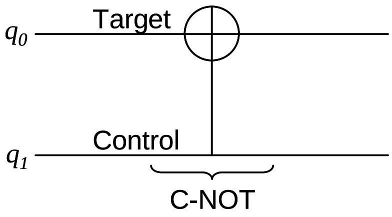


Figure 2-3: Quantum circuit representation of a C-NOT gate

2.1.3 Entanglement

Entanglement is, next to parallelism and interference, one of the special features of quantum computation. It can be understood by analysing following circuit:

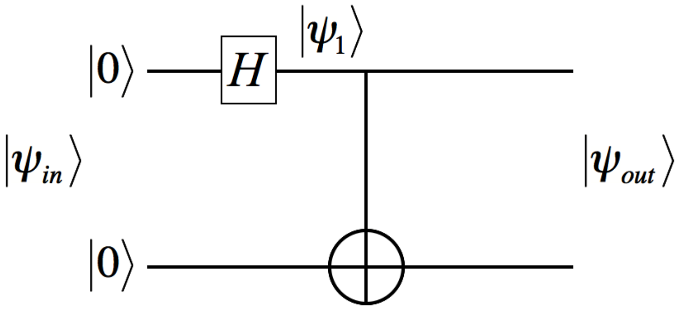


Figure 2-4: Entangling circuit

Supposing that the input state is:

$$|\psi_{in}\rangle = |00\rangle \tag{2.24}$$

after the Hadamard gate on qubit qb_0 one obtains:

$$|\psi_1\rangle = |0\rangle \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right) = \frac{1}{\sqrt{2}}\left(|00\rangle + |01\rangle\right) \tag{2.25}$$

Applying the C-NOT gate gives:

$$|\psi_{out}\rangle = \frac{1}{\sqrt{2}}\left(|00\rangle + |11\rangle\right) \tag{2.26}$$

This state is an entangled one since, as described, for example, in [Nie03, DiV00], mathematically, it is not possible to express $|\psi_{out}\rangle$ as some form of a product state

$$\left(\alpha|0\rangle + \beta|1\rangle\right)\left(\gamma|0\rangle + \delta|1\rangle\right) \tag{2.27}$$

of its two qubits. Computationally, this has following implications: if one measures the state of qubit qb_1, for example, expression (2.26) says that there is 50% probability that the superposition may collapse to the $|0\rangle$ state and 50% that it will collapse to the state $|1\rangle$. Supposing that the superposition after measuring the state of qubit qb_1 collapses to the state $|0\rangle$, the two-qubit register ends up in the state:

$$|\psi_{out}\rangle = |00\rangle \tag{2.28}$$

Hence, the measurement of qubit qb_1 has also influenced the state of qubit qb_0, which is a strong correlation of the qubits in a quantum register, a characteristic of quantum systems, which does not exist in the classical world.

2.2 Introduction to some experimental aspects of quantum computation

In the following sections, some mathematical notation describing the physics of quantum computational systems will be given. First, the qubit, and specifically spin-qubits, will be described in more detail. Then, decoherence, i.e. the unwanted interference of the environment with the quantum computational system [Nie03], will be discussed. Finally, the 5 criteria defined by DiVincenzo [DiV00], which any experimental quantum computational systems should satisfy will be described.

2.2.1 The spin-qubit

There are many different quantum physical systems, which can be used to store qubits for a quantum register and some of these will be reviewed in Chapter 3. However, because most of these systems are based on different types of well defined spin states, e.g. systems based on nuclear magnetic resonance [Chu98] or spin systems embedded in solid-state systems [Kan98, Los98], or on systems which can be easily described through similar tools, e.g. Josephson junction qubits [Mak01], a closer look at their mathematical description will be taken here. Further, because in the SFG quantum computation proposal qubits are carried by the spins of electrons, the discussion will be based on an electron spin-qubit, although very similar results hold for other spin systems.

Let us consider an electron placed in a static magnetic field B_z aligned in the z direction. The spin of the electron is then quantized along z [Bra03] and the parallel and anti-parallel orientation of the spin with respect to B_z can be used as a two-level quantum mechanical state for the storage of a qubit, for example, defining the parallel orientation as the $|0\rangle$-state and the anti-parallel as the $|1\rangle$-state.

The dynamics of such a system can by analysed solving the Schrödinger equation shown in expression (2.10) and reported here again with $\hbar = 1$:

$$i\frac{\partial|\psi(t)\rangle}{\partial t}=H_{Ham}|\psi(t)\rangle \tag{2.29}$$

Supposing that the qubit starts in the state $|0\rangle$ (spin-up, aligned with the static magnetic field) its state can be changed using electron spin resonance (ESR) by applying a.c. electromagnetic pulses [Bra03] orthogonal to the static magnetic field B_z whose interaction with the two-level quantum mechanical systems is described by the Hamiltonian:

$$H_{Ham}=-\vec{\mu}\cdot\vec{B} \tag{2.30}$$

where $\vec{\mu}$ is the magnetic dipole moment of the electron and $\vec{B}$ typically has the form:

$$\vec{B}=B_z\hat{z}+B_x\hat{x}\cos(\omega t) \tag{2.31}$$

with B_x the magnitude of the orthogonal a.c. field, usually several order of magnitudes smaller than B_z, and ω its frequency. For simplicity, it is assumed that the only contribution to the electron's angular momentum comes from its spin component[Bra03]:

$$\vec{S}=\frac{1}{2}\vec{\sigma} \tag{2.32}$$

where $\vec{\sigma}$ is a vector whose components are the Pauli spin matrices:

$$X=\begin{bmatrix}0 & 1\\ 1 & 0\end{bmatrix};\quad Y=\begin{bmatrix}0 & -i\\ i & 0\end{bmatrix};\quad Z=\begin{bmatrix}1 & 0\\ 0 & -1\end{bmatrix} \tag{2.33}$$

Under these assumptions, the Hamiltonian of this system can be expressed as:

$$H_{Ham}=\frac{g\mu_B}{2}\left[B_zZ+B_x\cos(\omega t)X\right] \tag{2.34}$$

where g is the Landé factor and μ_B the Bohr magneton. Defining $\omega_0=g\mu_BB_z$ and $A=g\mu_BB_x$, expression (2.34) becomes:

$$H_{Ham}=\frac{\omega_0}{2}Z+\frac{A}{2}\cos(\omega t)X \tag{2.35}$$

Inserting this Hamiltonian in the Schrödinger equation yields:

$$i\frac{\partial|\psi\rangle}{\partial t}=\left[\frac{\omega_0}{2}Z+\frac{A}{2}\cos(\omega t)X\right]|\psi\rangle \tag{2.36}$$

An analytical solution to this equation can be derived by shifting into a rotating frame through the substitution[Zha05]:

$$|\psi_1\rangle=e^{i\frac{\omega_0}{2}Z}|\psi\rangle \tag{2.37}$$

which allows to eliminate the terms in $\frac{\omega_0}{2}Z$ on the right hand side of expression . The resulting expression can be shown to depend on oscillating terms of the form $e^{\pm i(\omega+\omega_0)t}$ and $e^{\pm i(\omega-\omega_0)t}$. However, for ω~ω_0 the slow oscillating terms in ω - ω_0 will lead to a stronger contribution in the solution of the Schrödinger equation. Hence, an approximate solution can be obtained by neglecting the fast oscillating terms in $\omega + \omega_0$, through the so-called rotating wave approximation which returns:

$$i\frac{\partial|\psi\rangle}{\partial t}=\left[+\frac{A}{4}X+\frac{\omega-\omega_0}{2}Z\right]|\psi\rangle \tag{2.38}$$

In the rotating frame picture, expression (2.38) has solution[Zha05]:

$$|\psi(t)\rangle=e^{-i\left(\frac{A}{4}X+\frac{\omega_0-\omega}{2}Z\right)t}|\psi(0)\rangle \tag{2.39}$$

where $|\psi(0)\rangle$ is the state of the qubit prior to the application of the a.c. electromagnetic field.

In expression (2.39), when $\omega << \omega_0$ or in the absence of the a.c. electromagnetic, the *Z*-term is dominant and the transformation reduces to:

$$U \approx e^{-i\frac{\omega_0}{2}Zt} \tag{2.40}$$

which describes, in the Bloch sphere representation, a rotation of the spin state by an angle $\vartheta = \omega_0 t$ around the *z*-axis. Hence, when the static magnetic field is dominant, the spin precesses around the *z*-axis with frequency ω_0. When ω approaches ω_0, the spin's precession frequency around the *z*-axis is reduced to $\omega_0 - \omega$. At resonance, i.e. $\omega = \omega_0$, the static magnetic term becomes stationary and it is the a.c. electromagnetic field which dominates the interaction and rotates the spin around the *x*-axis with frequency *A/2*. Hence, by alternating periods of free evolution of the spin in the static magnetic field B_z (corresponding to rotations of the spin around the *z*-axis) with spin rotations around the *x*-axis induced by a.c. electromagnetic pulses centred on the resonance frequency ω_0, any point on the Bloch sphere can be reached. Computationally, this means that any single-qubit operation can be implemented by combining a static magnetic field with an a.c. electromagnetic signal with variable frequency and phase. ESR (and similarly nuclear magnetic resonance (NMR) for qubits stored in the nuclear spin of atoms [Nie03,Chu98,Cor97]) is, therefore, a commonly used technique for implementing single-qubit operations in quantum computational systems based on spin-qubits.

2.2.2 Decoherence

Ideally, one wants the qubits to be completely isolated from the environment or to be under the effect of some specific engineered interactions, such as the interaction with electromagnetic fields applied to change the state of the qubits in order to implement a desired quantum logic gate. However, in every experimental implementation of a quantum register there will be unwanted coupling mechanisms of the environment with the qubits – a phenomenon described as *decoherence*[Chu95,Unr95]. These unwanted interactions of the environment with the qubits can change their state in an unpredictable way and, hence, introduce errors in the computation. The specific decoherence mechanism which affects a quantum computation depends on the experimental set-up of the quantum computer, i.e. on the quantum mechanical system used to memorize the qubits and on the environment which surrounds the quantum register. It varies, therefore, from physical implementation to physical implementation. A useful parameter for the analysis of the impact of decoherence on a specific implementation of a quantum computer is the ratio n_{op} of the decoherence time τ_Q, i.e. the amount of time after which the impact of decoherence cannot be neglected anymore, and the gate operation time τ_{op}, i.e. the total amount of time necessary to apply a single quantum logic gate to the quantum register[Nie03]:

$$n_{op} = \frac{\tau_Q}{\tau_{op}} \tag{2.41}$$

The ratio n_{op} gives a rough estimate of the total number of quantum gates which one is able to apply to the quantum register before decoherence interferes with the qubits and should, therefore, be large enough to allow to implement the desired quantum computation[DiV00]. This leads to two possibilities. For short quantum algorithms, it may be sufficient that the total computation time is much shorter than the decoherence time, as shown, for example, in the experiment presented in [Chu98]. However, for more complex algorithms in which the computation time is expected to be longer than the decoherence time, errors introduced by decoherence (or by other mechanisms such as, for example, the non-optimal control of the quantum gates) may successfully be compensated through quantum error-correction codes and fault-tolerant quantum computation schemes[Sho95,Ste96,Sho96].

As mentioned before, decoherence mechanisms vary from implementation to implementation. However, in quantum systems which exploit spins for the qubits, the decoherence time τ_Q can be studied by analysing the transverse relaxation rate T_2, i.e. the rate with which the spin components transversal to an external magnetic field decay, and the longitudinal relaxation rate T_1, i.e. the rate with which a spin system placed in a magnetic field returns to thermal equilibrium once it has been displaced from it [Nie03, Hak87]. Computationally, T_2 describes the time-scale with which the relative phase difference of superpositions of words stored in a quantum register are perturbed by decoherence, while T_1 describes the time-scale over which an exchange of energy from the quantum register to the environment may take place which can lead to the unwanted relaxation of the excited states used for storing information in a qubit (e.g. $|1\rangle \rightarrow |0\rangle$)[Nie03]. In Chapter 3, values of decoherence times for different physical implementations of quantum computers will be discussed and compared.

2.2.3 The 5 DiVincenzo criteria

A variety of physical systems have been proposed for implementing quantum computation and all these systems differ in the way, for example, they store or manipulate the qubits or on how the qubits are made to interact. DiVincenzo[DiV00] defined a set of 5 criteria which any physical system proposed for quantum computation must satisfy for correct operation, some of which were at least partly already addressed above. These criteria have defined an important reference for understanding whether or not a proposed system is adequate for quantum computation, they help one to understand the main features of each implementation despite the different physical systems and to compare different implementations more easily.

The 5 criteria, or requirements, for quantum computation defined by DiVincenzo are the following [DiV00]:

1) A scalable physical system with well characterised qubits:

Any quantum computational system has as its main building block a collection of qubits, i.e. a collection of well characterised two-level quantum mechanical systems, used to store and process quantum information. As reviewed above, such a collection of N qubits can be used, thanks to superposition, to store simultaneously up to 2^N values, while entanglement provides strong correlation between the states of the qubits.

2) The ability to initialise the state of the qubits to a simple fiducial state:
Generally, to perform a mathematical computation, one needs to know the initial value of the information register. The same applies to quantum computation where, typically, this is achieved by initialising the quantum register to a known state. A state to which quantum registers are often initialised is the $\left|00....00\right\rangle$-state, as most of the times this corresponds to having all the qubits in their ground state. The initialisation can then be performed by letting the qubits relax to their ground state or by measuring the qubits and rotating them to the $\left|0\right\rangle$-state in case the output of the measurement is $\left|1\right\rangle$. Once, the register is in a well defined state, for example $\left|00....00\right\rangle$, the computation can begin by loading a desired value on the register by rotating the corresponding qubits to the $\left|1\right\rangle$-state, or by loading superpositions of states by applying Hadamard gates H to the qubits.

3) Long relevant decoherence times, much longer than the gate operation time:
From above, decoherence times give a measure of the unwanted coupling of the environment to the qubits. They describe the order of magnitude of time after which the influence of the environment on the quantum information stored in the quantum register cannot be neglected anymore and unpredictable errors are likely to occur. Again, while for short algorithms it is sufficient that the total computational time is shorter than the decoherence time, for the case of longer algorithms, in which the total computational time is longer than the decoherence time, the limits are defined by the necessity of implementing fault-tolerant quantum error correction codes and translate into having a decoherence times which allow one to operate with gate errors around 10^{-4} -10^{-6}.

4) A universal set of quantum gates:
As reviewed above, universality is the ability to compute any mathematical function with a computational system and translates in quantum computation to the ability of implementing arbitrary single-qubit rotations on all qubits and entangling gates between the qubits [Deu95,Div95,Nie03]. Experimentally, this means that any proposed physical

system needs to comprise ways of manipulating the states of the two-level quantum mechanical systems used as qubits and the entangling interactions between them.

5) A qubit-specific measurement capability

Once all the operations which implement a desired mathematical function have been applied, it is necessary to read-out the result stored in the qubits. This means that one has to be able to perform a measurement on any of the two-level quantum mechanical systems implementing the qubits in the quantum register.

As will be discussed in more detail in the next chapter, many implementations have been proposed and each of them approaches the five requirements in a different way. However, to date, few of the proposed implementations have managed to fulfil all 5 requirements experimentally, and none of these have proved to be scalable, i.e. able to connect and control any needed number of qubits. The most important example of an experimental quantum computation is probably still brought by the 7-qubit NMR factoring experiment presented in [Van01], while most of the remaining demonstrated approaches only managed to control quantum registers of few qubits. The question of which physical system will perform best in the long run and, especially, which one will prove scalable, is still open and is object of extensive research [Zol05].

2.3 Design of quantum logic circuits

A quantum computation is typically performed by applying a well-defined sequence of single- and two-qubit gates to the qubits of the quantum register[DiV00]. Hence, to design a quantum logic circuit means to find the sequence of single- and two-qubit gates which implements the desired computation. One way to approach this problem is by restating it in mathematical terms. A quantum algorithm can be described by a unitary matrix U_{comp} which, when applied to the state vector describing the quantum register, brings it into the desired final state. The matrix U_{comp} is an abstract object which does not define which single- and two-qubit gates to apply to the qubits, it only describes the transformation which the quantum register as a whole must undergo during computation. On the other hand, as mentioned above, each single- and two-qubit gate is itself described by a unitary matrix U_k. Hence, mathematically, given a unitary matrix U_{comp} describing a specific quantum algorithm and given a set of single- and two-qubit gates each described by a unitary matrix U_k (corresponding to the quantum transformations which a chosen technology is able to produce) the problem of designing the quantum logic circuit which implements the given algorithm corresponds to finding a decomposition of the matrix U_{comp} through the gates U_k. Or, in other words, to find a sequence of matrices U_k which, once multiplied together, will give U_{comp}. For example, consider a three-qubit quantum register. Further, suppose that the quantum computer one is using is able to produce three two-qubit gates U_1, U_2 and U_3 respectively between qubits 1 and 2, 2 and 3 and 1 and 3 and the single-qubit operations U_4, U_5 and U_6, respectively, on qubits 1, 2 and 3. Given a quantum algorithm described by the matrix U_{comp}, then a sequence of gates, for example, [U_4 U_1 U_3 U_6 U_2 U_6] is a quantum circuit which implements the given algorithm if:

$$U_{comp}=U_4 \cdot U_1 \cdot U_3 \cdot U_6 \cdot U_2 \cdot U_6 \qquad (2.42)$$

Or, using the wire-diagram representation described above:

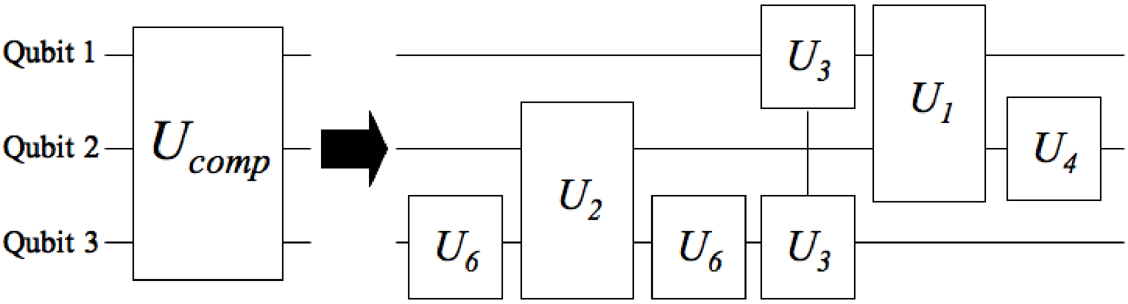


Figure 2-5: Quantum circuit design corresponds to decomposing the matrix U_{comp} into the product of matrices U_k which describe the available single- and two-qubit gates

Such decomposition, and therefore the quantum logic circuit which solves a given algorithm, is not unique. Firstly, different physical implementations of a quantum computer generate different sets of gates U_k available for the decomposition. A quantum logic circuit obtained for a given algorithm and for one type of quantum computer may differ from the one obtained for a quantum computer exploiting a different technology. Also, the design process may have to consider some physical constraints. Typically, the most important one is related to the impact of decoherence and requires, as described above, that this impairment is controlled either through quantum circuits which have computation time much shorter than the decoherence time or using quantum error correction codes and fault-tolerant computation schemes.

The main aim of the work presented in this thesis was to design quantum logic circuits suitable for the experimental demonstration of a small quantum computation prototype based on the SFG model. Hence, following constraints for this specific quantum circuit design problem were identified at the beginning of the project: Firstly, in order to test the behaviour of SFG gates in an experiment while keeping the technological challenges to the simplest possible level, the goal was set on developing compact quantum circuits with computation time shorter than the decoherence time. Secondly, the quantum gates available for the design of the circuits would only have been those which can be produced within the SFG model. This, however, still left a number of options open since, as will be shown in Chapter 4, there are different quantum gates which can be produced through the SFG model. It was not clear whether it would have been more convenient to use the SFG model for approximating gates commonly used in the literature such as the C-NOT gate, the controlled-phase gate or the $\sqrt{SWAP}$ gate (see, for example, [Pet05,Rod04]) or if more compact circuits may have been obtained using other gates which can be produced within the SFG technology.

The quantum circuits which are proposed in this thesis for the experimental demonstration of a small prototype quantum computer based on SFG gates will be presented in Chapter 6, while Chapters 3, 4 and 5 describe the work which helped identifying a number of systems parameters (e.g. the size of the quantum register, the quantum algorithm to be implemented by the circuits and the gates which can be produced within the SFG model) and the design techniques and tools used for deriving the circuits.

2.4 Summary

Quantum computers exploit two-level quantum mechanical systems, typically called qubits, as the basic elements for information storage and processing. These computational systems are characterised by 3 particular features: the superposition effects of quantum mechanical states (which is responsible for the implementation of parallel computation), the strong correlations between computational states introduced by entanglement and the phenomenon of interference of the information stored in the quantum register. As well summarised by DiVincenzo’s 5 criteria, any physical implementation proposed for quantum computation has to comprise:

1) A set of well defined qubits for storing quantum information

2) A known state to which the quantum register can be initialised at the beginning of the computation

3) Decoherence times of the qubits much longer than the total computational time or, when using quantum error correction codes and fault-tolerant quantum computation schemes, long enough to produce quantum gates with error rates around 10^{-4}-10^{-6}.

4) A universal set of quantum gates to implement the desired functions

5) A measurement mechanism for extracting the result of the computation from the quantum register

To implement a desired quantum computation one needs to design the corresponding quantum logic circuit which ultimately means finding the sequence of single- and two-qubit gates which brings the qubits of the quantum register to the desired final state. Such a sequence is typically not unique and the design process may often be bound by some physical constraints, often the most important one being the tolerance towards decoherence.

In the next chapter, the main characteristics of some of the most important quantum computation implementations will be reviewed, with the last section dedicated to solid-state implementations of spin-qubits since it is believed that, thanks to the knowledge and expertise acquired through classical electronics, these systems may have the potential for achieving large scalability.

2.5 References

[Bar95] A.Barenco, C.H.Bennett et al.: "Elementary gates for quantum computation", *Physical Review A.*, vol.52, pp.3457-3467, (1995)

[Bra03] B.H.Bransden, C.J.Joachain: "Physics of atoms and molecules", Prentice Hall, (2003)

[Chu95] I.L.Chuang, R.Laflamme, P.W.Shor, W.H.Zurek: "Quantum computers, factoring, and decoherence", *Science*, vol.270, pp.1633-1635, (1995)

[Chu98] I.L. Chuang, L.M.K. Vandersypen et al. : "Experimental realization of a quantum algorithm", *Nature*, vol. 393, pp.143-146, (1998)

[Cle98] R.Cleve, A.Ekert et al.: "Quantum algorithms revisited", *Proceedings of the Royal Society of London, Series A- Mathematical, Physical and Engineering Sciences*, vol.454, pp.339-354,(1998)

[Cor97] D. G. Cory, A. F. Fahmy et al.: "Ensemble quantum computing by NMR spectroscopy", *Proceedings of the National Academy of Sciences of the United States of America*, vol.94, pp.1634-1639, (1997)

[Deu85] D.Deutsch: "Quantum Theory, the Church-Turing Principle and the Universal Quantum Computer", *Proceedings of the Royal Society of London, Series A, Mathematical and Physical Sciences*, vol.400, pp.97-117, (1985)

[Deu89] D.Deutsch: "Quantum computational networks", *Proceedings of the Royal Society of London, Series A, Mathematical and Physical Sciences*, vol.425, pp.73-90, (1989)

[Deu92] D.Deutsch, R.Jozsa: "Rapid solution of problems by quantum computation", *Proceedings of the Royal Society of London, Series A, Mathematical and Physical Sciences*, vol.439, pp.553-558, (1992)

[Deu95] D.Deutsch, A.Barenco et al.: "Universality in quantum computation", *Proceedings of the Royal Society of London, Series A, Mathematical and Physical Sciences*, vol.449, pp.669-677, (1995)

[DiV00] D.P.DiVincenzo: "The physical implementation of quantum computation", *Fortschritte Der Physik*, vol.48, pp.771-783, (2000)

[DiV95] D.P.DiVincenzo, "Two-bit gates are universal for quantum computation", *Physical Review A*, vol.51, pp.1015-1022, (1995) [Hak87] H.Haken, H.C.Wolf: “Atomic and quantum physics”, Springer-Verlag, second enlarged edition, (1987)

[Kan98] B.E. Kane: “A silicon-based quantum computer”, *Nature*, vol.393, pp.133-137,(1998)

[Los98] D.Loss, D.P.DiVicenzo: “Quantum computation with quantum dots”, *Physical Review A*, vol.57, p.120-126, (1998)

[Mak01] Y.Makhlin, G.Schön et al.: “Quantum-state engineering with Josephson-junction devices”, *Review of Modern Physics*, vol.73, pp.357-400,(2001)

[Nie03] M.A. Nielsen, I.L.Chuang: “Quantum computation and quantum information”, Cambridge: Cambridge University Press, (2003)

[Pet05] J.R.Petta, A.C.Johnson et al.: “Coherent manipulations of coupled electron spins in semiconductor quantum dots”, *Science*, vol.309, p.2180-2184, (2005)

[Rod04] R. Rodriquez, A.J. Fisher, P.T.Greenland, A.M. Stoneham, “Avoiding entanglement loss when two-qubit quantum gates are controlled by electronic excitation”, *Journal of Physics: Condensed Matter*, vol.16, pp.2757-2772, (2004)

[Sch03] F. Schmidt-Kaler, H.Häffner et al.: “Realization of the Cirac-Zoller controlled-NOT quantum gate”, *Nature*, vol.422, pp.408-411, (2003)

[Sch95] B.Schumacher, “Quantum coding”, *Physical Review A*, vol.51, pp.2738-2747, (1995)

[Sha48] C.E.Shannon, “A mathematical theory of communication”, *The Bell System Technical Journal*, vol.27, pp.623-656, (1948)

[Sho95] P.W.Shor: “Scheme for reducing decoherence in quantum computer memory”, *Physical Review A*, vol.52, pp.R2493-R2496, (1995)

[Sho96] P.W.Shor: Fault-tolernat quantum computation”, *37th Annual Symposium on Foundations of Computer Science*, Book Series: Annual Symposium on Foundations of Computer Science, pp.56-65, (1996)

[Ste96] A.M.Steane: “Error correcting codes in quantum theory”, *Physical Review Letters*, vol.77, pp.793-797, (1996)

[Unr95] W.G.Unruh: “Maintaining coherence in quantum computers”, *Physical Review A*, vol.51, pp.992-997, (1995) [Van01] L.M.K.Vandersypen, M.Steffen et al.: “Experimental realization of Shor’s quantum factoring algorithm using nuclear magnetic resonance”, *Nature*, vol.414, pp. 883-887, (2001)

[Zha05] J.Zhang, K.B.Whaley: “Generation of quantum logic operations from physical Hamiltonians”, *Physical Review A*, vol.71, Article number 052317, (2005)

[Zol05] P.Zoller, T.Beth et al.: “Quantum information processing and communication - Strategic report on current status, visions and goals for research in Europe”, *European Physical Journal D*, vol.36, pp.203-228, (2005)

Chapter 3 Physical implementations of quantum computers

As described in the Introduction, although the concept of quantum computational systems was developed more than 20 years ago, it was the discovery made by Shor in 1994 of a quantum factoring algorithm and its potential for breaking the widely used cryptographic system RSA[Sho94], that really drew the broad interest from the scientific research community to the area of quantum computation. Since then, much effort has been invested in finding quantum-mechanical systems able to store qubits and ways of making these qubits interact in order to perform quantum computations. Or, in other words and remembering the requirements defined by DiVincenzo[DiV00] reviewed in the previous chapter, the rush to find the most convenient physical system able to initialise, store, process and read-out quantum information was triggered by Shor's discovery. Hence, as previously mentioned, there are many different implementations of quantum computers which have been proposed and even, at least partly, demonstrated experimentally. Nevertheless, large-scale quantum computation has not been achieved yet and it is not clear which system might prove best[Zol05].

As will be described in Chapter 4, the SFG proposal is based on optically controlled, solid-state quantum logic gates. It aims at achieving scalability exploiting the knowledge and expertise of solid-state systems acquired from classical electronics, their promising resilience towards decoherence and the optical control of the two-qubit interactions which will allow one to avoid high-precision fabrication techniques for the placement of control electrodes used, for example, in Kane's proposal[Kan98]. One important step for testing the potential of these gates is to analyse their behaviour in a small-scale experiment and proposing suitable quantum circuits for such an experiment was the main goal of the work presented in this thesis. Hence, the first step of this work required to review the status of experimental quantum computation in order to understand what it means to prepare the experimental demonstration of a prototype system and to identify system parameters such as, for example, convenient quantum algorithms or the typical size of quantum registers.

The main results of this review are given in this chapter. It starts with an historical analysis aimed at understanding the main challenges and objectives of experimental quantum computation focusing on NMR, ion-traps and Josephson-junction qubit systems. The final part deals with implementations based on spin-qubits in solid-state systems because of their promising features for the development of scalable quantum computational systems and focuses on quantum dots, spin-qubits bound to donors in silicon, diamond systems and NMR in silicon.

3.1 First steps in experimental quantum computation

3.1.1 Nuclear magnetic resonance

The first experimental demonstrations of a complete quantum algorithm have been performed on quantum computational systems based on NMR[Cor97,Ger97]. In an NMR quantum computer, the quantum register consists of qubits stored in the nuclear spins of atoms within the same molecule. Single-qubit operations are performed by applying selective radiofrequency electromagnetic pulses which change the spin state of a chosen nucleus whereas two-qubit operations can be implemented, as will be described further, thanks to coupling mechanisms between the nuclei of the molecule[Ger97].

Exploiting this system, the experimental demonstration of quantum algorithms was started through two two-qubit versions of Deutsch's problem[Deu85,Deu92,Cle98] which were presented by Chuang et al. and Jones et al., respectively, in [Chu98] and [Jon98]. In [Kim00] a revised Deutsch-Jozsa algorithm[Col98] on 3 qubits was demonstrated. Then, an experimental demonstration of an order-finding algorithm was implemented[Van00] which finally led to the implementation of Shor's factoring algorithm[Sho94] on a 7-qubit NMR quantum computer[Van01], one of the milestones of experimental quantum computation.

To date, a large number of experiments have been performed on NMR based quantum computers. Only recently, for example, two new experiments implementing three-qubit Deutsch-Jozsa algorithms have been presented[Fah08,Gop08]. While there are examples of experimentally demonstrated quantum registers comprising more than 10 qubits, e.g. 12 in [Neg06], most of the reported experiments have demonstrated systems of 2 to 4 qubits.

3.1.1.1 Description of the NMR experimental setup

Figure 3-1 shows a typical experimental setup needed for an NMR quantum computation[Nie03]. A molecule used for NMR quantum computation usually contains a number n of atoms with spin ½ nuclei which store the qubits. Each nucleus represents an independent qubit of the quantum register with, for example, the information '0' coded to the spin of the nucleus being aligned parallel to a static magnetic field B_0 and the information '1' to the spin being anti-parallel to the static magnetic field. The first state is defined as the $|0\rangle$-state and the latter as the $|1\rangle$-state. Moreover, each nucleus is characterised by a different *Larmor* frequency ω_{0i} [Van04] (i.e. the frequency with which a spin precesses around the z-axis defined by the static magnetic field B_0) in order, as will be described in more detail below, to be able to selectively address the qubits and for measuring their state. Because of the small magnitude of the nuclear magnetic moment of a single spin, a large number of molecules (~10^{18} in [Chu98], for example) must be present in order to be able to read-out the result of the computation. Each molecule then represents a single and independent quantum computer. The molecules are dissolved in a solvent in order to make inter-molecular interactions negligible. The final system works as an *ensemble* of quantum computers with an output signal being the average of all the signals of the different molecules.

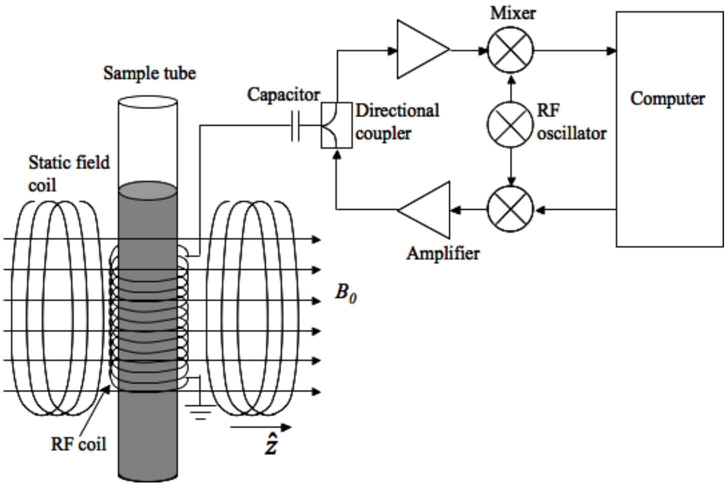


Figure 3-1: Schematic of an NMR quantum computer experimental setup[Nie03]

As shown in Figure 3-1 the sample tube containing the solution is placed in the bore of the static $\hat{z}$-oriented magnetic field B_0 which, as mentioned above, sets a reference direction for the spins. Radiofrequency (RF) coils are placed around the sample tube and allow one to apply RF pulses in the $\hat{x}$ and $\hat{y}$ directions for manipulating the spin states of the qubits and, hence, to perform a computation. The same coils are used during the read-out phase for measuring the result of the computation and all operations are coordinated by a classical computer.

As described in Section 2.2.1 of Chapter 2, a single qubit rotation is applied to a spin-qubit by transmitting an RF pulse centred on the resonance frequency ω_{0i} of the corresponding nuclear spin[Nie03]. Depending on the length of the RF pulses and on the orientation ($\hat{x}$ and $\hat{y}$, for example) of the magnetic field vector, rotations to any point of the Bloch sphere can be obtained using this technique. Molecules used in NMR quantum computers with more than one qubit are built in such a way that each nucleus has a different resonance frequency ω_{0i} in order to be able to selectively address the single qubits. Typical resonance frequencies of molecules used in NMR quantum computation experiments are in the range of ~100s of MHz[Chu98].

Two-qubit gates are implemented in NMR quantum computation by exploiting an interaction between adjacent nuclear spins mediated by the electrons shared in a chemical bond. As mentioned in Section 2.2.1, the precession frequency ω_{0i} of a spin is proportional to the strength of the magnetic field it is immersed in. In the setup described in Figure 3-1, the magnetic field experienced by a nuclear spin will not only be influenced by the static magnetic field B_0, but also by the state of the electronic cloud of the nucleus. When this electronic cloud overlaps with the electronic cloud of another nucleus, an interaction of the two nuclei mediated by the two electronic clouds is established which can lead to a perturbation of the local magnetic field of one nucleus dependant on the state of the other one[Nie03]. The local perturbation of the magnetic field experienced by the nucleus causes a shift in its Larmor frequency. Hence, the effect of this interaction is a shift in the Larmor frequency of one nucleus conditional to the state of the other nucleus it is interacting with. The strength of this interaction is described by the parameter J which quantifies the resulting frequency shift[Van04]:

$$\omega_{0iJ} = \omega_{0i} \pm \frac{(2 \cdot \pi \cdot J)}{2} \tag{3.1}$$

'-' for spin j in state 'up'

'+' for spin j in state 'down'

where ω_{0iJ} is the shifted Larmor frequency. This shift in the precession frequency of a nucleus dependent on the state of another nucleus can be used, for example, to implement a C-NOT gate between the two adjacent qubits *i* and *j* by transmitting a narrowband pulse centred on frequency $\omega = \omega_{0i} + \frac{2 \cdot \pi \cdot J}{2}$ [Ste01]. As can be seen from expression (3.1), spin *i* will be flipped by this pulse only if pulse *j* is in the 'down' (i.e. $|1\rangle$) state, which is equivalent to a C-NOT gate.

Typical values of *J* can be in the range of about 1 to hundreds of Hz[Nie03,Van01]. *J* also gives an estimate on the duration of the two-qubit gate. As described above, a C-NOT gate can be implemented by transmitting a narrowband pulse. To select the desired frequency, for example $\omega = \omega_{0i} + \frac{2 \cdot \pi \cdot J}{2}$, without exciting the transition resonant to $\omega_{0i} - \frac{2 \cdot \pi \cdot J}{2}$, the transmitted pulse must be characterised by a bandwidth of the order of *J* which corresponds to pulse widths of the order of *1/J*. From this point of view large values of *J* give short gate operation times τ_{op} and help to keep the total computation time short.

Finally, the result of the computation can be obtained by measuring the Larmor frequencies of the nuclei since, as described above, these are subject to a shift which depends on the state of their neighbours. Hence, by knowing the Larmor frequency of each nucleus it is possible to reconstruct the value stored by each qubit. This can be achieved by transmitting an RF reference signal on the sample and by sweeping the magnetic field B_0. By changing the magnetic field B_0, the energy splittings of the nuclei are changed and, therefore, their Larmor frequencies. Whenever B_0 reaches a value which changes the energy of one of the nuclei such that its Larmor frequency coincides with the reference RF signal, the system is brought to resonance and the RF signal is absorbed by the sample. Hence, by analysing the absorption of the reference RF signal as a function of the value of B_0, it is possible to evaluate the Larmor frequencies of the nuclei and, therefore, the values stored by each qubit.

3.1.1.2 Limitations in NMR quantum computation

NMR quantum computation is an ensemble computation, i.e. the measurement of the final signal gives the average of all the results coming from each molecule of the solution, each of which acts as an independent quantum computer. Hence, the average of the results coming from an ensemble of quantum computers may be a number which is uncorrelated with the correct solution of the problem. However, as is the case, for example, of the factoring algorithm, this problem can be solved by means of some classical post-processing performed directly on the quantum register which is always possible since quantum computation subsumes classical information processing[Nie03].

In terms of decoherence, typical T_2 rates reported in NMR quantum computation experiments are of the order of ~ s [Van01,Van00]. Although relatively long, this has to be compared with J values between 1 and 100s of Hz which correspond to two-qubit gate operation times between 1ms to 1s, which makes the implementation of algorithms requiring more than a few tens of two-qubit gates difficult.

Another important limitation of NMR quantum computation comes from the computation being applied to an ensemble of quantum computers. As described before, in an NMR quantum computation experiment a large number of molecules is necessary in order to obtain a measurable signal. Each molecule represents an independent quantum computer. At the beginning of the experiment, which usually is performed at room temperature, the solution will be in the thermal equilibrium state, meaning that, in first approximation, each nucleus (qubit) of each molecule (quantum computer) can be considered as having 50% of probability of being in the ‘up’ ‘or’ down’ state. This means that the solution containing all the quantum computers will comprise some registers starting in the $|00...00\rangle$ state, others in the $|00...01\rangle$, and so on, with an almost equal distribution of population between those states. The quantum system is then said to be in a *mixture* of the pure states $|00...00\rangle, |00...01\rangle .... |11...11\rangle$, while in reality one wants the quantum register to be in a well defined state, typically the $|00...00\rangle$ state [DiV00]. Although there are techniques for extracting a so called *effective pure state* this happens at the expense of extra overhead required for the computation. An *n*-qubit initial state $|00...00\rangle$ can be obtained from a quantum register of $q>n$ qubits, for example [Nie03]. The problem of these techniques is that the final signal decays exponentially with the number n of qubits distilled into effective pure states[Nie03], limiting the scalability of such a quantum computation system. NMR quantum computation systems are believed to be scalable to some tens of qubits[Ste01].

Because of these limitations, NMR quantum computation is unlikely to be one of the candidates which will implement large-scale quantum computation. Nevertheless, it is the implementation which has allowed to make the initial steps and define the main problems in experimental quantum computation.

3.1.2 Ion Traps

Quantum computation with ion traps was first proposed by Cirac and Zoller in [Cir95], where it was suggested that trapped ions could be sufficiently well isolated from the environment to create quantum systems with long decoherence times which would be suitable for quantum computation.

As described in [Cir95], the idea is to store a qubit in two internal states of an ion. A quantum register of N qubits is created by trapping N ions combining the effect of an electromagnetic field and the repulsive Coulomb force of the ions. The trap is designed such that the ions form a linear string. Interactions between the qubits are induced through the collective motion of the string of ions in the trap and are controlled through optical pulses. From the computational point of view, the collective motion (or motional state) of the string of ions can be seen as a further qubit which interacts with all the ion-qubits of the string. The first experimental implementation of a C-NOT gate in an ion trap was demonstrated in [Mon95] using one $^9Be^+$ ion. The C-NOT gate was applied to the qubit memorized in the two internal states of the ion and the qubit represented by the motional state of the ion in the trap. Then, a Deutsch-Jozsa algorithm on two qubits and a C-NOT gate on two ions were demonstrated experimentally, respectively, in [Gul03] and [Sch03a]. In [Sch03a] the qubit represented by the motional state of the two ions was used as an information "bus" between the two qubits encoded in the internal states of the ions. Also, other studies have concentrated on the investigation of new entangling techniques[Sør00,Lei03] and to novel ion trap designs oriented at enhancing the scalability of the system [Kie02]. While quantum registers of up to 6 qubits have been reported in literature[Lei05], typical system sizes of recent experiments comprise quantum registers of 2 and 3 qubits[Ben08,Mon09].

3.1.2.1 Description of the ion-trap experimental setup

As summarized in [Nie03] and schematized in Figure 3-2 the ions are trapped by an electromagnetic field (comprising both a static and an oscillating part) applied through four cylindrical electrodes. Together with the Coulomb repulsion of the ions this system can be seen as an harmonic oscillator in which the ions oscillate at frequencies ω_x, ω_y and ω_z along the axes. Further, the trap is designed in such a way that ω_x, $\omega_y >> \omega_z$ in order to have the possibility to bias the system such that the ions only lie and oscillate along the *z*-axis. If the trap is sufficiently well isolated from the environment then the motion of the electromagnetically confined ions becomes quantized. The number of normal modes on which the chain of ions can oscillate is proportional to the number of ions in the trap and, for the one-dimensional case, it is equal to *N*[Hug98]. In this regime, the eigenstates of this harmonic oscillator represent motional states of the entire string of ions moving together as one system. The first energy levels of this harmonic oscillator are spaced in units of $\hbar\omega_z$ and each quantum of motional energy $\hbar\omega_z$ is called a phonon. A specific normal mode can be populated with any number of phonons since these are bosons which obey a Bose-Einstein statistics[Bra03]. For a given motional mode, its phonon state will here be labelled as $\left|n = 0,1,2...\right\rangle$ where *n* stands for the number of phonons. As mentioned before, in ion-trap quantum computation the motional states of the string of ions are used to induce interactions between the ion-qubits. The motional levels which are typically exploited for computation are the motional ground state, i.e. $\left|n = 0\right\rangle$ in which each ion rests around its equilibrium position, and the centre-of-mass (COM) mode with $\left|n = 1\right\rangle$ in which the entire chain of ions oscillates like a single body along the *z*-axis[Cir95]. These two motional levels can be treated computationally as a qubit linked to all the ion-qubits of the string and can therefore be used to process and exchange information between them[Sch03a]. Quantum gates are applied to the quantum register of ion-qubits through optical pulses while photodetectors are used, as will be described later, during read-out.

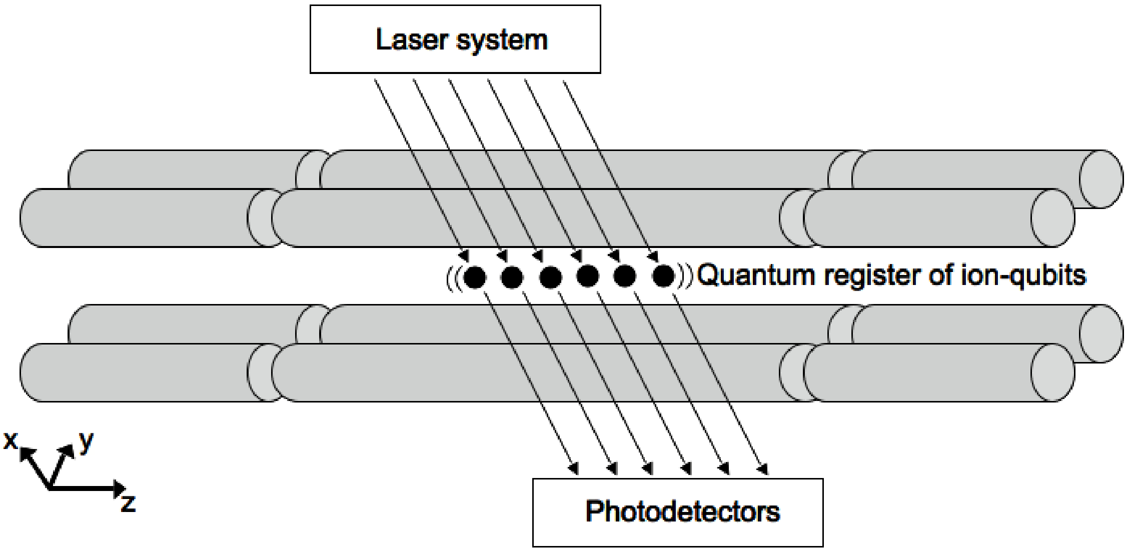


Figure 3-2: Schematic of an ion trap[Nie03]

To understand the main features of ion-trap quantum computation it is useful to analyse the energy levels of an ion-trap qubit. The qubit is implemented in some internal atomic states of the ions in the trap. Each ion memorizes a different qubit. Generally, the $|0\rangle$ state is encoded in the ground state of the ion whereas the $|1\rangle$ state is encoded in some excited state which can be accessed optically. Because, as described before, the ions are linked by their motional state, it is important also to keep track of the number of phonons characterising the states. Hence, the state of a single ion-qubit in the trap will be labelled through two pairs of integers: $|0,1;n=0,1,2,..\rangle$. The first one indicates the logical value memorised by the qubit, the actual quantum information, while the second describes the amount of phonons in the trap. Figure 3-3 describes the energy states of a single ion-qubit in the trap[Chi00]. The ground state is the $|0;n=0\rangle=|00\rangle$ state. The system can be cooled down to this state by means of optical cooling techniques in which states with many phonons $|0;n\rangle$ are optically excited to the $|1;n-1\rangle$ state which has a high probability of relaxing to the state $|0;n\rangle$, $|0;n-1\rangle$ or $|0;n-2\rangle$. Hence, by repeating this cycle a sufficient amount of times, the system will end-up in the ground state $|00\rangle$[Nie03].

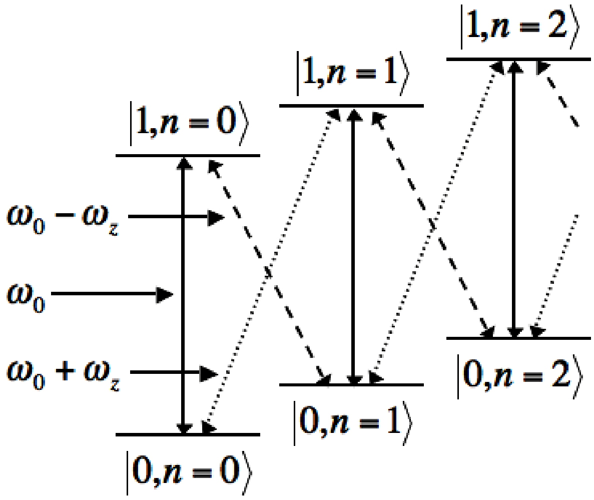


Figure 3-3: Energy levels and possible transitions for one ion[Chi00]

A single qubit operation on an ion can be implemented by transmitting an optical pulse centred on frequency ω_0. Depending on the length of the pulse, a qubit starting from the logical $|0\rangle$ state can be excited to the $|1\rangle$ state or to any superposition of the two [Chi00,Gul03].

In terms of two-qubit gates, many different proposals have been developed since Cirac and Zoller's first idea[Cir95]. However, we will focus on one of the earlier developments which led to the first successful demonstration of a two-qubit gate applied between two ion-qubits in a trap [Sch03a]. From Figure 3-3, it can be seen that the logical states of an ion trap qubit ($|0\rangle$ and $|1\rangle$) can interact with the motional state ($|n=0\rangle$ and $|n=1\rangle$) of the entire string of ions if it is excited with an optical signal centred on $\omega_0 \pm \omega_Z$. If, for example, the string of ions is in the motional ground state $|n=0\rangle$ and one of the ions is in the logical state $|0\rangle$ (hence globally being in the state $|0,n=0\rangle$), then it is possible to bring the ion to the logical state $|1\rangle$ and set the entire string in motion by bringing it to the state $|n=1\rangle$ by exciting the ion through an optical pulse centred on $\omega_0+\omega_Z$. Computationally, this mechanism can be used to transfer the information stored in the ion-qubit to the motional state. Once the information has been transferred to the motional state, because the entire string of qubits is affected by the motional state, it can be made to interact with any other qubit in the register exploiting, again, the transitions centred on $\omega_0 \pm \omega_Z$. Hence, to apply a two-qubit gate between ion-qubits A and B, first, the information of qubit A is transferred to the motional state of the string of ions, then, an interaction between qubit B and the motional state is implemented. The protocol ends by transferring the information stored in the motional state back to ion A. Using this scheme, the first C-NOT gate between two ion-qubits was demonstrated[Sch03a].

Read-out of the information stored in ion trap qubits is achieved exploiting an auxiliary level $\left|aux_{read-out}\right\rangle$ which couples efficiently only with one of the two internal states, for example state $\left|0\right\rangle$ as shown schematically in Figure 3-4, where, for simplicity, the motional energy levels have been omitted. When a laser tuned on the transition between the $\left|0\right\rangle$ and the $\left|aux_{read-out}\right\rangle$ state is shone on one of the ions, then fluorescence light is collected through photodetectors only if the ion was in the $\left|0\right\rangle$ state.

This method is very powerful because one measurement can be performed inducing thousands of transition cycles between the $\left|0\right\rangle$ and the $\left|aux_{read-out}\right\rangle$ state allowing one to accumulate good statistics.

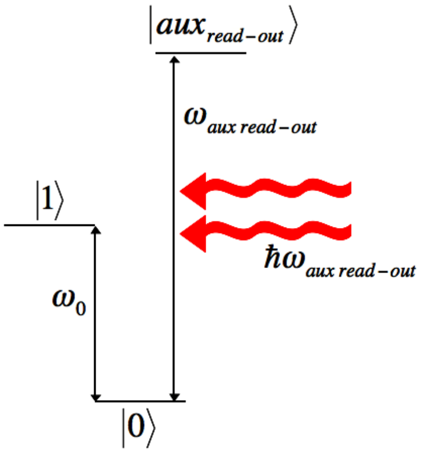


Figure 3-4: Auxiliary level exploited for read-out. The $\hbar\omega_{aux\text{read-out}}$ radiation only couples with the $\left|0\right\rangle$ state. If this state is populated, fluorescence will be collected by the photodetectors of the system

3.1.2.2 Limitations in ion-trap quantum computation

To date, one factor which seems to be limiting the precision of ion-trap quantum gates is the non-optimal control over experimental parameters (such as fluctuations in the optical sources, for example), although it is expected that these problems should be addressable in the future[Bla08].

In terms of decoherence times, as described in [Haf08], values typically reported in ion-trap quantum computation experiments are in the order of ~ms with gate operation times in the area of hundreds of μs, although longer decoherence times may be achievable.

Focusing on the issue of scalability, it seems difficult to trap more than some tens of ions[Kie02,Wei03], since the addition of each ion adds three motional modes in the trap, making it difficult to spectrally isolate the desired motional states. Alternative architectures have been presented in which the quantum computer is composed by a set of linked ion traps with a small number of ions. Ions from different sub-traps are then "shuttled" into an interaction region in which multi-qubit operations are performed. This architecture is called "quantum charge-coupled device" architecture and was presented in [Kie02].

3.1.2.3 Neutral atom traps

As described above, ions can be tightly confined in ion traps and their manipulation has been demonstrated experimentally although exploiting the trapping potential in conjunction with the Coulomb repulsion limits the number of ions which can be precisely controlled in a trap. Also, because ions are charged, they tend to couple with noisy electromagnetic fields of the environment. Because of these drawbacks, it became clear, that some of these problems could be solved if the ions were replaced by neutral atoms trapped through intersecting laser beams[Bre99]. However, while neutral atoms experience a lower coupling with the surrounding environment, they also tend not to couple between themselves, making the introduction of two-qubit interactions the critical element of neutral atom quantum computation[Jes04]. To date, single qubit manipulation of trapped neutral atoms has been demonstrated experimentally, as reported, for example, in [Sch04]. While quantum operations on arrays of neutral atoms qubits have been demonstrated, it is the issue of implementing two-qubit gates on selected qubits within an array which still has to be solved[And07].

3.1.3 Josephson junction qubits

Josephson junctions are superconducting devices which have been proposed, for example in [Shn97], as viable systems for storing qubits and implementing single- and multi-qubit quantum gates. The first physical implementation of a Josephson junction qubit was demonstrated in [Nak99], which led then to the development of experimental two-qubit gates as shown, for example, in [Yam03] and, more recently, in [Pla07]. One of the strengths of this proposal is believed to lie in the fact that Josephson junction qubits can be fabricated by established lithographic methods, which might enhance the scalability of such systems.

3.1.3.1 Description of the setup

As shown in [Mak01], there are different proposals for quantum registers based on Josephson junctions which differ in the way single qubits are implemented, in how qubits are coupled together and they are expected, in general, to have different tolerances towards decoherence. Here, the model presented in Figure 3-5[Mak99,Mak01] has been chosen to describe Josephson Junction quantum computation because it describes well its main features and characteristics.

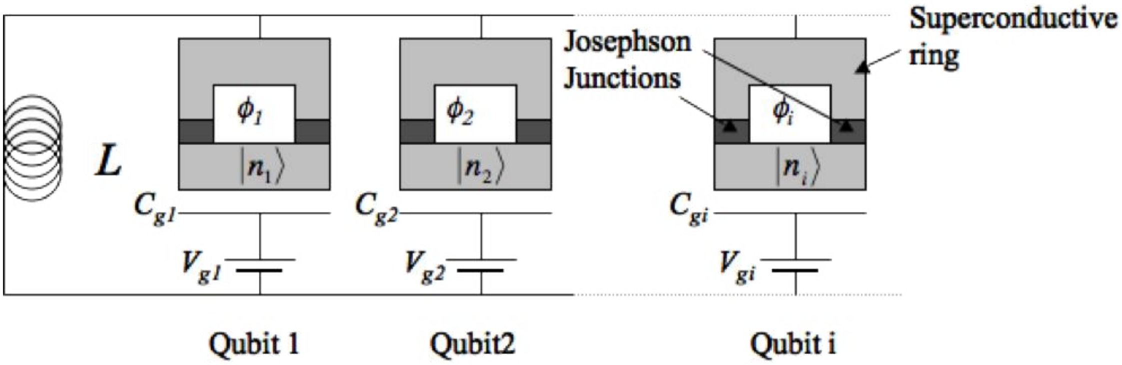


Figure 3-5: Experimental set-up for a Josephson Junction quantum register[Mak99,Mak01]

Each qubit is implemented in one of the subsystems consisting of the two superconductive islands which form a ring only separated by the two Josephson junctions represented by the capacitive junctions C_j, forming a device called superconductive quantum interference device (SQUID). The SQUID is coupled through a gate capacitor C_{gi} to a voltage source V_{gi} and with the magnetic flux ϕ_i passing through it. In the superconductive regime electrons pair up in so-called Cooper pairs [Buc04], which can tunnel through the Josephson junction from one superconductive island to the other. Through the gate voltage V_{gi} and the magnetic flux ϕ_i it is then possible to control the number n_i of excess Cooper pairs in the lower superconducting island, forming a system typically called a Cooper pair box[Mak01]. By adjusting these two parameters it is also possible to bias each Josephson junction qubit to an *idle* state in which the qubit does not "see" the rest of the circuit and does therefore not interact with it. The qubit is formed by taking the $|n_i = 0\rangle$ state of the Cooper pair box as the quantum logical state $|0\rangle$ whereas the $|n_i = 1\rangle$ state corresponds to the state $|1\rangle$. Single qubit operations are applied by controlling these states through the tunnelling in and out of the superconducting island of a Cooper pair. As described in [Mak01], this model is equivalent to a spin-qubit when associating the spin-up state to the sate $|n_i = 0\rangle$ and the $|n_i = 1\rangle$ state to spin-down. In this picture, the gate voltage V_{gi} and the flux component ϕ_i have, respectively, an effect equivalent to two time-varying magnetic fields aligned along z and x and can therefore be used, as in NMR quantum computation, to bring the Josephson Junction qubit to any superposition of its states $|n_i = 0\rangle$ and the $|n_i = 1\rangle$.

By observing Figure 3-5, it can be seen that the capacitive parts of the quantum register and the inductance L form a common LC circuit. This is exploited to apply two-qubit operations. Firstly, suppose that all qubits are set to the idle state. Then, the two qubits which have to interact are biased out of the idle state. As described in [Mak01], the common LC circuit introduces coupling between the two qubits which leads to interactions equivalent to the ones seen in the previous implementations and can therefore be used to obtain the same two-qubit gates. Once the desired two-qubit gate has been implemented, the two qubits are brought back to the idle state where they stop interacting with each other.

The state of each qubit can be read-out by measuring the amount of charge inside each Cooper-pair box. Experimentally, this has been demonstrated in [Nak99,Yam03] and is shown schematically in Figure 3-6. Each Cooper-pair box is connected to a probe junction which is biased through a voltage source V_b in such a way, that the excited state$|n_i = 1\rangle$, will decay to the ground state $|n_i = 0\rangle$ with two tunnelling events through the probe-junction[Nak99]. Hence, the state of the qubit can be determined by measuring the current through the probe junction. If a current is measured, the state of the qubit was $|1\rangle$, otherwise it was $|0\rangle$. The drawback of this method is that the junction is always connected to the Cooper-pair box representing a constant source of decoherence which, in [Nak99], was recognized as being the dominant decohering effect.

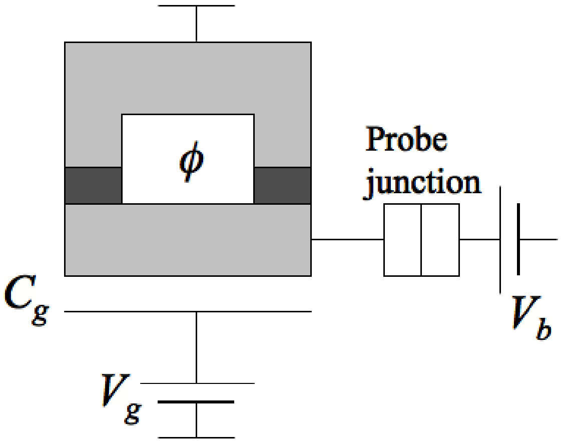


Figure 3-6: Probe junction for read-out

Hence, a new measurement mechanism has been proposed which aims at coupling the measurement device to the qubit only during the actual measuring phase, while it is biased to a state less decohering for the qubit during quantum manipulations. As reviewed, for example in [Mak01], this can be achieved by connecting the gate of a single-electron transistor (SET) to the Cooper pair box forming the qubit, introducing on the current flowing through the SET a dependence on the charge state of the qubit. However, when a measurement is not needed, it is possible to bias the SET to a state in which it mainly changes the capacitance in the system, reducing its decohering impact on the qubit.

3.1.3.2 Limitations in Josephson junction quantum computation

Josephson junction qubits have been extensively studied during the last years and are believed to be a very promising system for quantum computation. However, as summarised in [Wen07], improvements in, both, decoherence times and scalability, will be necessary if these devices are to be used in quantum computation processors. As reviewed in [Cla08], decoherence times have been brought from the ~ns range of the first experiments up to ~µs in more recent demonstrations. In terms of scalability, to the best of our knowledge, only quantum manipulations between two qubits have been demonstrated experimentally as shown, for example in [Pla07,Yam03], making the implementation of quantum gates on registers of at least 3 qubits one of the important short-term goals for the development of quantum computation with Josephson junction qubits[Wen07].

3.2 Spin-qubits in solid state systems

We have analyzed some of the implementations of quantum computers which started the experimental research in this area in order to understand its main features and problems. While NMR and ion-trap quantum computers have allowed to collect precious experimental evidence of the fundamentals of quantum computation it has been shown that that they are not scalable to more than a few tens of qubits [Ste01,Kie02,Wei03], although, for the case of ion-traps, alternative trap architectures may lead to scalable systems. As reviewed in [Cla08], quantum computers based on Josephson junction qubits seem to be a promising candidate and many studies are dedicated to this research area although more work still needs to be done to assess their potential for large scale computation. Table 3-1 summarises some of the main experimental parameters for the quantum computational systems reviewed in this first part of the chapter.

Table 3-1: Typical experimental parameters for NMR, ion trap and Josephson junction quantum computational systems

	Typical system size	Decoherence	Typical gate operation time	Potential for scalability
NMR	2-4 qubits	~s	~10-100ms	No
Ion traps	2-3	~ms	~100μs	Possible, with novel trap structures
Josephson junctions	2	~μs	~10ns	Possible

As reviewed previously, although quantum registers of, respectively, 12 and 6 qubits have been demonstrated for NMR and ion-trap quantum computers, many of the presented experiments still focus on system comprising 2-4 qubits. For NMR systems, for example, two of the most recently presented experiments demonstrated three-qubit versions of the Deutsch-Jozsa algorithm[Gop08,Fah08], while, for the case of ion-traps, a three-qubit Toffoli gate represents one of the latest demonstrations[Mon09]. Further, decoherence times longer than the gate operation times have been demonstrated, although only some tens to hundreds of operations are typically feasible to date during the decoherence time.

Hence, due to the exceptional technological challenges involved in experimental quantum computation (which concern the impact of decoherence as well as the precise control of the instrumentation used for manipulating the qubits), the road towards building a scalable system is still long[Zol05]. This is also reflected in the many different proposals for quantum computers which have been developed during the last years. Particularly, a lot of interest is being given to qubits implemented in spin-states of solid-state systems as the expertise and fabrication techniques acquired through classical electronics may prove helpful for achieving greater scalability[Cer05, Das05]. Also, measurement of the decoherence times of some spin-qubits have shown to be very promising[Tyr03]. The following sections will focus on quantum dots[Los98], on Kane's proposal[Kan98] and on some alternative schemes of nuclear and electron spin qubits in solid-state systems[Cha01,Lad02,Sto03].

3.2.1 Quantum dots

Quantum dots are semiconductor devices based on heterostructures in which it is possible to control the number of confined electrons down to a single one[Ash96]. As in atoms, the energy levels of the confined electrons in quantum dots are quantised, making them systems in which it is possible to study and control quantum phenomena[Kou98]. Quantum dots were brought to the attention of the quantum information processing community when Loss and DiVincenzo published a proposal for a quantum computer based on coupled quantum dots[Los98]. This proposal triggered extensive research in the area of quantum computation which finally led to the experimental demonstration of electrically controlled single- and two-qubit operations[Kop06,Pet05], read-out and initialisation[Elz04]. These studies also showed that ensemble decoherence times T_2* of around ~10ns could be expected while spin-echo analysis indicated that T_2 decoherence times of ~1μs can be obtained[Pet05]. Also, proposals for optically controlled quantum dots are being developed, as demonstrated, for example, in [Pre08], which can operate faster compared to electrically controlled quantum dots, and much work is being invested in assessing their potential[Cer05].

3.2.1.1 Description of the set-up

Figure 3-7 shows a schematic of a quantum dot three-qubit register[Gol02].

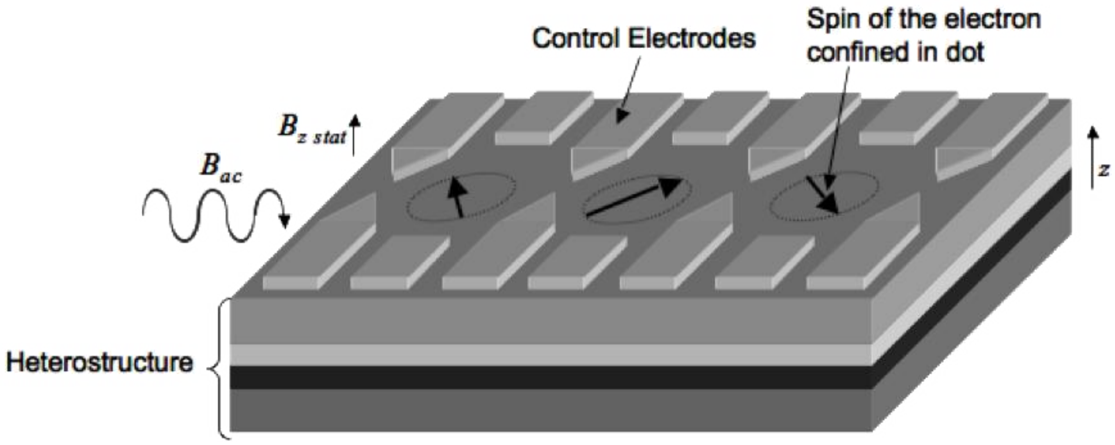


Figure 3-7: Schematic of a quantum dot qubit register. The heterostructure creates a 2DEG parallel to the chip surface. By charging the control electrodes negatively it is possible to deplete the 2DEG until a single electron is trapped between the gates.

The heterostructure creates strong confinement for electrons along the z-direction in Figure 3-7, but allows them to move freely in the plane perpendicular to z. Such a system is called a 2 dimensional electron gas (2DEG). By charging the control electrodes negatively it is possible to deplete the 2DEG until a single electron remains confined between them. Figure 3-7 represents schematically a quantum register with three quantum dots and one electron confined in each dot. The static magnetic field $B_{z\,stat}$ sets the reference direction for the spins, while the RF magnetic field B_{ac} is used for single qubit operations. The electrodes on the chip are also used for controlling the interaction between adjacent dots and for reading-out the spin state of the electrons.

The quantum dot qubit is implemented in the electron trapped by the potential minimum created by the negatively charged electrodes. As schematically described in Figure 3-8(a)[Elz04], the undepleted area close to the dot acts as an electron reservoir from which electrons can tunnel into the dot or from the dot into the reservoir depending on the bias of the control electrodes on the chip. Because of the tight confinement of the electron in the dot, the energy levels of such a system are quantized. In order to reduce the impact of decoherence due to thermal energy, quantum dots have to be kept at very low temperatures (~100mK [Pet05]). The spin-up and spin-down states of the energy ground-state of the confined electron are then the two chosen levels for the quantum dot qubit as schematized in Figure 3-8(b):

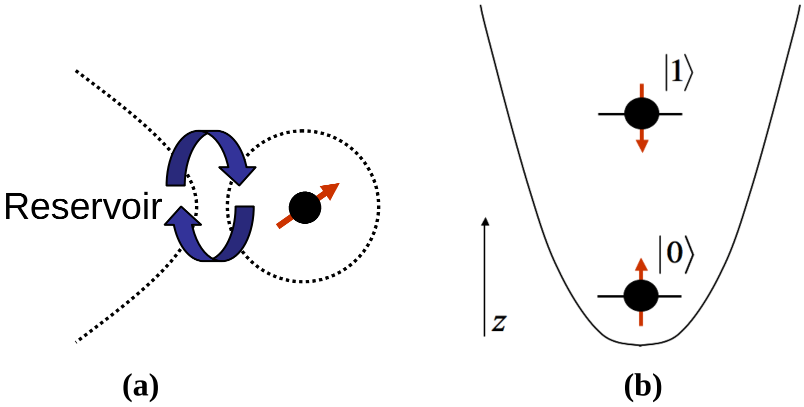


Figure 3-8:(a) Schematic representation of a quantum dot and its reservoir. Electrons can be made to tunnel into or out of the dot from the reservoir by proper tuning of the control electrodes[Elz04]. (b) Schematic representation of the quantum dot spin-qubit and its energy levels. The electron is trapped in the potential minimum. An external magnetic field along *z* sets the reference for the qubit states.

As reviewed in the second chapter, transitions between the spin-up and spin-down state, and, hence, single qubit operations, can be induced by transmitting RF magnetic pulses B_{ac} resonant with the electron-spin precession frequency in the static magnetic field B_{zstat} and orthogonal to it. Such a scheme has been demonstrated experimentally in [Kop06].

Interactions between adjacent quantum dot qubits are mediated by controlling the potential barriers generated by the electrodes on the chip. A schematic for the two-qubit case is shown in Figure 3-9:

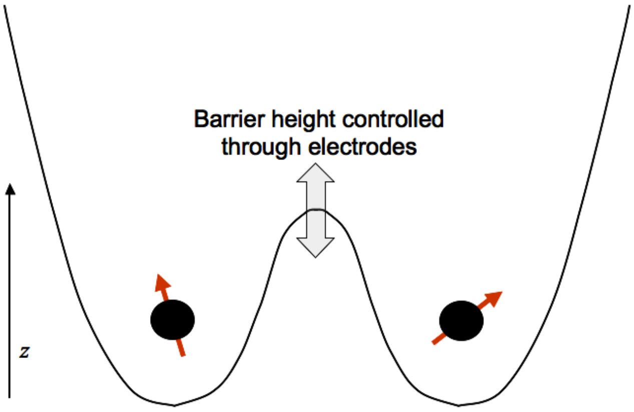


Figure 3-9: Interaction between two adjacent quantum dot qubits. For high barriers, the two electrons are isolated from each other. If the barrier is lowered it is possible to introduce enough overlap between the electrons' wavefunction to have interaction between the two.

Through the control electrodes it is possible to keep the potential barrier between the dots so high, that tunnelling between the two dots is inhibited and the two electrons are isolated from each other. However, if by changing the bias voltage on the chip electrodes, the potential barrier is lowered such that tunnelling of the electrons between the dots is allowed, then overlap between the electrons' wavefunction can be obtained which corresponds to an effective interaction between the qubits[Cer05]. Using this scheme, interaction between two electrons in two quantum dots and a square-root of swap gate have been demonstrated experimentally in [Pet05].

Read-out of the spin-state of an electron in a quantum dot can be performed by a spin-to-charge conversion and has been demonstrated experimentally by Elzermann et al.[Elz04]. The electron in the dot is coupled electrostatically to its environment and influences therefore the currents which flow in proximity of the dot[Kou06]. By monitoring such currents through a dedicated electrode system (typically termed quantum point contact (QPC) [Pet05]), it is possible to establish whether or not an electron is inside the quantum dot. Hence, to exploit the QPC for reading-out a quantum dot qubit, it is necessary to bind during the measurement procedure the presence of the electron inside the dot to its spin state which can be achieved using following protocol:

A) During the computation, the electrodes are biased such that the electron is trapped inside the dot.

B) The measurement starts by lowering the barrier (Figure 3-8(b)) of the dot such that only the spin-down state will tunnel into the reservoir while the spin-up state will be kept trapped inside the dot.

C) By monitoring the current through the QPC, which depends on the presence of the electron inside the dot, it is possible to reconstruct the state of the electron at the beginning of the measurement. The spin state of the electron is therefore determined by a measurement of the charge state of the dot.

As demonstrated by Elzerman et al.[Elz04], initialisation of a quantum dot qubit register can be obtained by filling each dot with a randomly polarised electron and waiting for relaxation of all spin-down states to the spin-up state.

3.2.1.2 Decoherence and scalability

In terms of decoherence, experimental measurements have returned values around 10-20ns[Pet05,Kop06]. This seems problematic considering that the length-scale of the single-qubit manipulations demonstrated in [Kop06] was of the same order of magnitude, although in [Pet05] this decoherence value was extended to ~1μs using spin-echo techniques. In both experiments, the main contribution to decoherence was recognized as being introduced by a fluctuating magnetic field generated by the nuclei of the atoms in the bulk which perturbs the quantum states of the qubits.

In terms of scalability, to the best of our knowledge, only quantum manipulations in quantum registers comprising two quantum dots have been demonstrated to date[Pet05,Kop06]. The creation and generation of entangled states between two or more spins is one of the topics this research area will shift to during the next years[Han08].

3.2.2 The Kane proposal

In 1998, B.E. Kane developed the proposal for a quantum computer with qubits carried by the nuclear spin states of phosphorus atoms embedded in a silicon substrate[Kan98]. Together with the Loss-DiVincenzo proposal for quantum dot quantum computation, this proposal shared the idea of exploiting decades of expertise in semiconductor technology for implementing a scalable quantum computer. The strength of these ideas, the potential for long coherence times of donor spin states and the theoretical and experimental know-how in the area of semiconductors deriving from the classic microelectronics field, rapidly attracted huge interest in the quantum computation community. New proposals were developed based on electron spin rather than nuclear spin qubits as reviewed in [Hog03], experiments were performed which showed that diamond is an interesting alternative to silicon substrates for spin-based quantum computation [Cha01] and a new quantum computer model which combined the successes of NMR quantum computation with an all-silicon quantum computer was proposed[Lad02]. Also, it has been demonstrated experimentally that the electron spin decoherence times for phosphorus-doped silicon are well in excess of ~1ms [Tyr03].

3.2.2.1 Description of the set-up

A schematic of Kane's proposal is shown in Figure 3-10[Kan98]. The qubits are implemented in the nuclear spin state of phosphorus atoms embedded in the silicon chip. Convenient separations between the phosphorus atoms are estimated to lie around 10-20nm while the system is kept at a temperature of ~100mK. ^{31}P is used as it has nuclear spin I=1/2 and because, being a shallow donor, its electron wavefunction can spread from tens to hundreds of Ångstroms from the nucleus. As will be described later, this feature is exploited for obtaining two-qubit interactions. Further, the electrons highlighted in Figure 3-10 also play an important role when implementing single-qubit operations and during the read-out procedure.

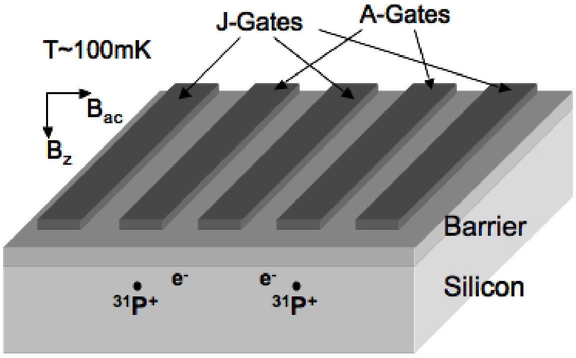


Figure 3-10: Schematic of the quantum register in the Kane proposal[Kan98].

A static magnetic field B_z sets the reference for the spin states while the a.c. magnetic field B_{ac} is used to flip the nuclear spins of the donors. Typically, the nuclear spin state parallel to the static magnetic field is chosen as the $|0\rangle$ state for the qubit, while the anti-parallel one for $|1\rangle$. There are two types of electrodes on the chip. The ones termed "A-Gates" are used to control the single-qubit operations. These are placed over the donor sites and allow, through appropriate bias as schematised in Figure 3-11, to move the wavefunction of the electron away from the nucleus. The state of the nucleus, and with that its resonance frequency, is influenced by the state of the electronic wavefunction and by moving the electron away from the nucleus, the strength of this interaction is reduced and, through that, the resonance frequency of the nuclear spin. To apply a one-qubit gate to a chosen qubit, the A-Gate electrode placed over the corresponding donor is biased such that the resonance frequency of the nuclear spin coincides with the frequency of the a.c. magnetic field B_{ac}, bringing the nuclear spin into resonance.

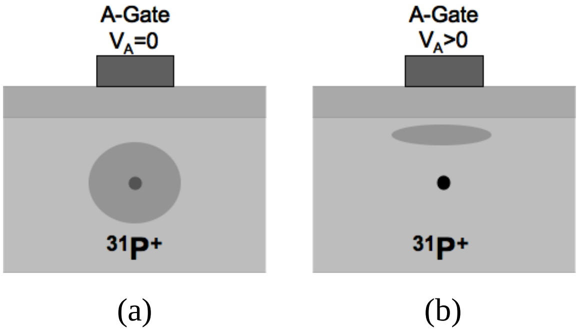


Figure 3-11: By tuning the A-gate over the desired donor it is possible to control the strength of the interaction between nucleus and the corresponding electron and, with that, the resonance frequency of the nuclear spin. In (a), for V_A=0, the wavefunction is distributed around the nucleus. (b) For V_A>0 the wavefunction is pushed towards the barrier and away from the nucleus, reducing the strength of the interaction.

Conversely, the "J-Gates" are used for introducing interactions between adjacent qubits. Again, the nucleus is influenced by the state of the electronic wavefunction. Further, two adjacent electrons can interact with each other through the overlap of their wavefunction. Two nuclei can therefore "feel" each other through the overlap of their electrons. In the Kane proposal, this electron-mediated interaction can be controlled with an appropriate bias of the J-gate electrode placed between the nuclei since this electrode influences the overlap of the electron wavefunctions, as shown schematically in Figure 3-12.

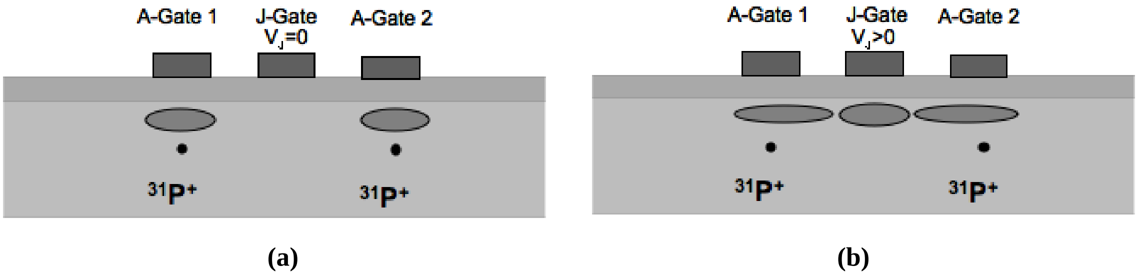


Figure 3-12: Two-qubit interaction mechanism. In (a), for V_J=0, the two electron wavefunctions are concentrated in proximity of the corresponding P donors. There is no overlap of the electron's wavefunction with the adjacent nucleus. The two nuclei do not interact. In (b), the positively charged J-Gate spreads the wavefunctions. The two nuclei "feel" each other through the overlap of their corresponding electron wavefunction

The read-out mechanism is based on two steps. First, the information stored by the nuclear spin of the qubit to be measured is transferred to the electron spin, then it is extracted through current measurements which depend on the state of the electrons. This is achieved with the help of an auxiliary qubit. As described in [Kan98], during computation the system is biased through the J-gate potential such that the information carried by two adjacent nuclear spins, i.e. $|00\rangle, |01\rangle, |10\rangle, |11\rangle$, is coupled to the $|\downarrow\downarrow\rangle$ state (parallel to the static magnetic field) of the shared electron pair. Instead, assuming the second qubit now to be the auxiliary qubit used for reading out the first one, for large *J* the system can be biased such that the $|1\rangle$ state of the qubit to be measured stays coupled to the $|\downarrow\downarrow\rangle$ state of the electrons while the state $|0\rangle$ couples to the state $|\uparrow\downarrow - \downarrow\uparrow\rangle$ independently from the state of the auxiliary qubit. Since only two electrons in the $|\uparrow\downarrow - \downarrow\uparrow\rangle$ state will bind to the same phosphorus atom, an electron transfer between the two qubits can be induced, and the corresponding current be measured, if their electrons are in the $|\uparrow\downarrow - \downarrow\uparrow\rangle$-state and, therefore, the qubit to be measured is in the state $|0\rangle$. Hence, the measurement is performed by coupling an auxiliary qubit to the one to be measured and by controlling whether or not an electron transfer between the two qubits can be induced.

Finally, initialisation can be achieved by cooling the system down to its operation temperature of ~100mK and by waiting for it to relax to its lowest energy level which corresponds to the quantum register being in the $|00\ldots00\rangle$ state.

3.2.2.2 Decoherence and scalability

Recent experiments with phosphorus-doped silicon crystals have demonstrated that decoherence times are in the range of tens of ms (with single-qubit gate operation times around ~20μs)[Mor08], confirming the potentials of qubits carried by the nuclear spin of phosphorus atoms embedded in silicon.

However, much work still needs to be done to implement the quantum computation system envisaged by Kane. To the best of our knowledge, no selectively controllable single- or two-qubit quantum register (including its read-out scheme) has been experimentally demonstrated to date. One of the problems seems to be the high precision required in the fabrication process of the quantum register due to the variation in the strength of the interaction of the donors which strongly depends on their positioning[Das05]. Nevertheless, very important results, such as, for example, the incorporation of phosphorus atoms in silicon with atomic-scale precision[Sch03b], have been achieved during the last years.

3.2.3 Developments and alternatives in solid-state quantum computation

Although, as reviewed above, many of the key features of Kane's proposal still have to be demonstrated experimentally, his proposal triggered great interest[Hog03] and new systems, inspired by his ideas, have been developed.

3.2.3.1 Electrons vs nuclear spins

Soon after Kane' s proposal, Vrijen et al. proposed an evolution of his idea in which the qubits were carried by the electron spin (rather than the nuclear spin) of a phosphorus atom embedded in a silicon substrate[Vri00], thereby avoiding the transfer of information from the nuclear to the electron spin required in Kane's model. Except for the medium carrying the qubit, the main principles of the proposal developed by Vrijen et al. are very similar to Kane's: single- and two-qubit interactions are controlled by influencing the state of the electron's wavefunction. Although, to the best of our knowledge, no fully operating few-qubit quantum register has been demonstrated to date, there is great interest for donor electron spin qubits, interest which was also enhanced by the measurements of decoherence times of several ms reported by Tyryshkin et al[Tyr03].

3.2.3.2 Nitrogen-vacancy centre in diamond systems

Quantum computation based on nitrogen-vacancy centres in diamond has attracted much interest during the last years through some very promising experimental demonstrations. In [Jel04a], for example, a qubit structure was demonstrated which exhibited decoherence times of ~1μs. But while this figure is of the same order of magnitude of the dephasing times demonstrated for quantum dots [Pet05], it was measured at room temperature, whereas quantum dots or the Kane proposal require temperatures down to fractions of Kelvin [Kan98,Pet05].

The main building block for such systems is the nitrogen-vacancy centre in a diamond crystal, i.e. a nitrogen atom substituting for a carbon atom in proximity of a vacancy as schematized in Figure 3-13(a)[Wra06]. The blue particles represent the carbon atoms, the red one is nitrogen while the grey one denotes the vacancy. Typically, a qubit is carried by the ground state of the two unpaired electrons of the nitrogen vacancy which form a spin triplet[Niz05], and are therefore characterised by S=1, where S is the quantum number associated with the magnitude of the total spin[Bra03]. The lowest energy level is formed by the triplet state with m_s=0 (where m_s is the quantum number associated to the z-component of the spin operator) which is usually chosen as the $|0\rangle$ state while the $|1\rangle$ state is stored in the two remaining states with $m_s = \pm 1$ which are degenerate and separated from the m_s=0 state by ~2.88GHz, as schematized in the energy level scheme in Figure 3-13(b)[Wra01,Niz05].

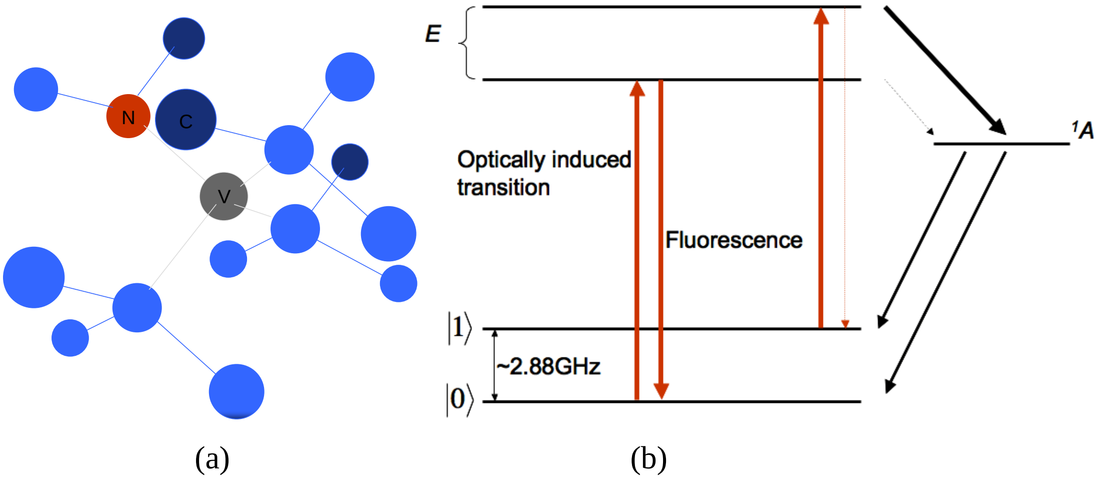


Figure 3-13:(a) NV structure[Wra06]. (b) Energy levels for single qubit operations, initialisation and read out[Wra01,Niz05].

One of the strengths of the nitrogen-vacancy proposal lies in the fact that, thanks to its energy level scheme, it allows to conveniently implement qubit manipulation, initialisation and read-out. As can be seen from Figure 3-13, the $|0\rangle$ and $|1\rangle$ states are separated by roughly 2.88GHz and can, therefore, be manipulated through microwave signals, while the separation between the ground and first excited state corresponds to 637nm which can be accessed optically and used, as described below, for read-out and initialisation of the qubits.

From Figure 3-13(b) it can be seen that, when stimulated optically, the $|0\rangle$ state tends to oscillate up and down between the ground and excited state, a behaviour which can be detected through its fluorescence signal. The $|1\rangle$ state, instead, once excited, may also relax down to the ground state via a third state ${}^{1}A$. This mechanism can be used for read-out since a strong fluorescence signal will only be measured if the qubit starts in the $|0\rangle$ state[Cha01,Jel02]. A fluorescence signal will also be measured when the qubit starts in the $|1\rangle$ state, but the system will eventually relax to the ${}^{1}A$ and the fluorescence signal will stop. Further, this behaviour can also be exploited to initialise the qubit to the $|0\rangle$ state. If the qubit is in the $|0\rangle$ state and it is optically pumped then it will return with high probability to its starting state. On the contrary, once excited, the $|1\rangle$ will relax via the ${}^{1}A$ state, either to the $|0\rangle$ state (and with that it will have completed initialisation) or to the $|1\rangle$ state, in which case the procedure must be repeated. Hence, by pumping this system optically long enough, the qubit will end in the $|0\rangle$ state with a high probability[Cha01]. Single-qubit manipulation, initialisation and read-out have already been demonstrated experimentally (as shown, for example, in [Cha01,Jel02, Jel04a]). In [Jel04a], typical single-qubit gate operation times were around some tens of ns.

In terms of two-qubit gates, interactions between the qubit described above and qubits carried by the nuclear spin of atoms placed in proximity of the nitrogen-vacancy centre have been demonstrated already, for example, in [Jel04b]. However, to the best of our knowledge, no interactions between qubits belonging to different nitrogen-vacancy centres have been demonstrated experimentally yet, which is key for system scalability.

Nevertheless, the great potential of this implementation has been recently further highlighted by room-temperature measurements which have demonstrated decoherence times in the order of ~ms[Bal09] (obtained using diamonds purified from unwanted carbon isotopes with non-zero nuclear spin which may perturb the coherence of the electron-spin qubit) and much attention is currently being given to this technology[Pra08, Han09].

3.2.3.3 NMR in silicon

In [Lad02], Ladd et al. proposed an all-silicon quantum computational system which combines solid-state quantum computation with traditional NMR. The basic building block of this implementation is shown in Figure 3-14[Ito05]:

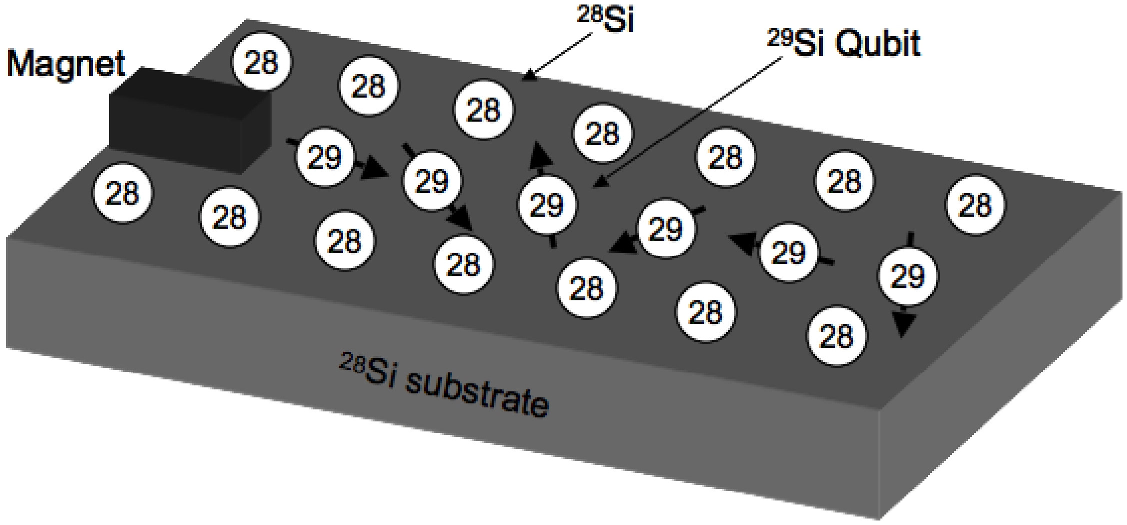


Figure 3-14: Basic building block of the silicon NMR quantum computer. Qubits are carried by the nuclear spin state of ^{29}Si while the substrate is made out of ^{28}Si atoms which have 0 nuclear spin[Ito05].

In the all-silicon NMR quantum computer, qubits are carried by the nuclear spin states of ^{29}Si atoms which have nuclear spin ½ while the substrate comprises purified silicon which mainly contains ^{28}Si atoms. These have nuclear spin 0 and, therefore, do not contribute to fluctuations of the local magnetic field around the qubits which would reduce the coherence time of the system[Lad02]. Quantum logic operations are controlled, as in standard NMR quantum computation, through RF pulses transmitted on the chip[Itoh05]. Selectivity of the single qubits is guaranteed by the static magnet placed at one end of the quantum register. This generates a variable magnetic field along the register which changes the precession frequency of each qubit and, hence, the RF frequency to which they react. Despite encouraging room-temperature decoherence times of the ^{29}Si qubits of the order of ~25s[Itoh05], no recent advances on the experimental development of such systems have been reported in literature to the best of our knowledge.

3.3 Summary

Some of the most important experimental implementations of quantum computation have been reviewed in this chapter. The aim of this review was to understand the state of the art of experimental quantum computation in order to comprehend the main challenges and the typical features of an experimental demonstration of a prototype quantum computational system. There are many quantum computation proposals which have been presented over the years and have not been described here (e.g. quantum computation based on the interaction of photons[Chu95] or on cavity quantum electrodynamics[Tur95]), since taking into account all of them would have been outside the scope of this work. Here, in the first part of the analysis, the focus has been set on NMR, ion-traps and Josephson-junction qubits because, when the research project presented in this thesis started, these were some of the technologies for which substantial results on the realization or development of a small-scale quantum computation systems had been obtained[Van01,Sch03a,Yam03].

In terms of system parameters demonstrated with these implementations, while registers of up to 12 qubits have been demonstrated, more frequent sizes of quantum registers are in the range of 2 to 4 qubits. In terms of algorithms, although the implementation of Shor’s factoring algorithm is often cited as one of the long-term aims of quantum computation (and has been implemented experimentally on a 7-qubit quantum register), a popular algorithm for the experimental demonstration of prototype quantum computational systems is the Deutsch-Jozsa algorithm. Its popularity is probably given by the fact that, despite its simplicity, it allows one to demonstrate the main features of quantum computation (i.e. parallelism, entanglement and interference) as long as the quantum register on which it is implemented is of at least 3 qubits[Col98,Nie03].

While it has been shown that NMR quantum computer are not scalable to more than a few tens of qubits, scalable systems may still be achieved using architectures based on ion-traps or Josephson-junctions and much work is being carried out in this area[Zol05].

The second part of the review focused on solid-state spin-qubit implementations because of their potential for high scalability[Cer05,Hog03]. Quantum dots[Los98] and the Kane proposal[Kan98] of qubits carried by the nuclear spin of phosphorus atoms embedded in silicon have been described and some alternative schemes which further developed Kane's idea (qubits implemented in the electron spin rather than the nuclear spin of phosphorus atoms[Vri00, Tyr03], nitrogen-vacancy systems in diamond[Cha01,Wra01] and all silicon NMR quantum computation[Lad02]) have been addressed. To date, some of these implementations have achieved the control of few-qubits quantum registers, although in most cases the focus of experiments is still on demonstrating specific tasks of the DiVincenzo check list rather than a complete demonstration through the implementation of proper algorithms. It is useful to have a look at the number of operations n_{op} which may be performed during the decoherence time of qubits implemented with different technologies. Rough estimates of this figure can be obtained by the ratio of the decoherence time T_2 with the gate operation time τ_{op} of a given physical implementation[Nie03]:

$$n_{op} \sim \frac{T_2}{\tau_{op}} \tag{3.2}$$

Table 3-2 summarises estimates of n_{op} for the set-ups discussed in this review, both, for single-qubit operations (SQO) and two-qubit operations (TQO), as well as typically reported operational temperatures.

Table 3-2: Estimates of the number of operations which can be implemented during the coherence time.

	n_{op} **(SQO)**	n_{op} **(TQO)**	**Operational temperature**
Diamond	~$4.5\cdot10^4$ [Jel04a,Pra08,Bal09] (1/n_{op}~$2.2\cdot10^{-5}$)	10^3 [Jel04b] (1/n_{op}~10^{-3})	Many experiments performed at room temperature (e.g. [Jel04a,Bal09])
Quantum Dots	~1-20 [Pet05,Kop06] (1/n_{op}~0.05-1)	7000 [Pet05] (1/n_{op}~$1.4\cdot10^{-4}$)	~100mK [Pet05]
Silicon NMR	10^6 [Ito05] (1/n_{op}~10^{-6})	10^4 [Ito05] (estimated, not yet experimentally proven) (1/n_{op}~10^{-4})	~5K [Ito05]
Phosphorus/Silicon (nuclear spin qubit)	~10^5 [Kan98,Mor08] (1/n_{op}~10^{-5})	~10^5 [Kan98,Mor08] (estimated, not yet experimentally proven) (1/n_{op}~10^{-5})	~100mK [Kan98]
Phosphorus/Silicon (electron spin qubits)	~$5\cdot10^4$ [Tyr03] (1/n_{op}~$2\cdot10^{-5}$)	~$5\cdot10^7$ [Tyr03, Das05], (estimated, not yet experimentally proven) (1/n_{op}~$2\cdot10^{-8}$)	Decoherence measurements performed at T~7K [Tyr03]

In terms of temperature of operation it should be noticed that most of the listed implementations require temperatures around a few K, sometimes down to fractions of K. The exception is the diamond nitrogen-vacancy system, which may have the potential for room-temperature quantum information processing[Bal09].

In terms of n_{op}, the values given in Table 3-2 should really only be considered as rough estimates, particularly the column concerning two-qubit operations; firstly, because in some cases (when specified), these estimates have been derived also on the basis of theoretical analysis with the experimental demonstration of the two-qubit interactions having yet to be implemented. Secondly, as seen in the review, none of the described systems has yet proved scalable and it may well be that their further developments may rely on different interaction mechanisms with different gate operation times. Nevertheless, Table 3-2 is useful to give the general idea of what the various implementations may achieve. For example, to crack a modern RSA cryptographic system with Shor's factoring algorithm[Sho94] one would need about 6000 qubits and 10^8 operations [Ger05]. It becomes clear looking at Table 3-2 that none of the listed physical systems is close to implementing such a large-scale computation. To harness such complex and time-consuming algorithms, fault-tolerant quantum error-correction, seems to be a necessary ingredient towards scalable quantum computation and its physical and scalable implementation is one of the main challenges research in the area of quantum information processing will have to face[Zol05]. In this context, one of the challenges is to produce quantum gates with acceptable error rate for fault-tolerant quantum computation. A typical reference value used as acceptable error rate is $\sim10^{-4}$ (see [Bla08,Cla08,Han09], for example) and a very rough estimate of the error rate produced by a certain technology can be obtained through the value $\frac{1}{n_{op}}$, assuming that one will perform n_{op} operations before an error will occur because of decoherence. From Table 3-2 it can be seen that various of the listed of implementations show values of $\frac{1}{n_{op}}$ which (at least in theory) are compatible with an error rate of $\sim10^{-4}$ and it will be important to see which of these implementations will be able to implement a scalable quantum register while maintaining acceptable error rates. However, to date, the road towards large-scale quantum computation seems still long and, although as can be seen from Table 3-2 some implementations seem more promising than others, the question of which system will finally prove best is still unanswered[Zol05].

However, there are applications which do not require the demanding resources of the factoring algorithm but would nevertheless accomplish very useful tasks. As mentioned in Chapter 1, one of these is quantum simulations, i.e. the modelling of quantum physical systems by a quantum computer[Lly96]. Using quantum registers of some tens of qubits would already allow to simulate systems intractable on a classical computer[Zol05,Nie03]. Thanks to this less demanding resources, quantum simulations could be one of the applications which may ultimately drive research, and possibly industry, in the near future [Spi06]. Hence, the implementation of quantum registers comprising a few tens of qubits represents a very important goal for exploiting the increased computational power of quantum computers.

In the next chapter, the SFG proposal will be described. As seen in the review here presented, although there are various systems suitable for storing qubits, one of the big challenges of quantum computation is finding interaction mechanisms for the qubits which do not limit the scalability of the system. Both, in Kane's and in the diamond proposal, for example, the qubits show promising decoherence times, but it is the interaction between the qubits which is proving difficult. The SFG proposal addresses this problem mediating the interaction of two adjacent donor electron spin qubits through a control particle placed in their proximity. The interaction mechanism between the qubits depends on the optical excitation of the control particle and can therefore be implemented without control electrodes, avoiding the high-precision fabrication techniques required in Kane's proposal. Further, the SFG model may also be compatible with the diamond scheme and could represent a way for introducing interactions between qubits stored in the nitrogen-vacancy centres[Sto09].

3.4 References

[And07] M.Anderlini, P.J.Lee, B.L.Brown et al.: “Controlled exchange interaction between pairs of neutral atoms in an optical lattice”, *Nature*, vol.448, pp.452-456,(2007)

[Ash96] R.C.Ashoori: “Electrons in artificial atoms”, *Nature*, vol.379, p.413-419, (1996)

[Bal09] G.Balasubramanian, P.Neumann, D.Twitchen, M.Markham et al.: “Ultralong spin coherence time in isotopically engineered diamond”, *Nature Materials*, vol.8, pp. 383-387, (2009)

[Ben08] J.Benhelm, G.Kirchmair, C.F.Roos, R.Blatt: “Towards fault-tolerant quantum computing with trapped ions”, *Nature Physics*, vol.4, pp.463-466, (2008)

[Bla08] R.Blatt, D.Wineland: “Entangled states of trapped atomic atoms”, *Nature*, vol.453, pp.1008-1015, (2008)

[Bra03] B.H.Bransden, C.J.Joachain: “Physics of atoms and molecules”, Prentice Hall, (2003)

[Bre99] G.K.Brennen, C.M.Caves et al.: “Quantum Logic Gates in Optical Lattices”, *Physical Review Letters*, vol.82 (5), pp.1060-1063,(1999)

[Buc04] W.Buckel, R.Kleiner: “Superconductivity Fundamentals and Applications”, Wiley, (2004)

[Cer05] V.Cerletti, W.A.Coish et al.:“Recipes for spin-based quantum computing”, *Nanotechnology*, vol.16, R27-R49, (2005)

[Cha01] F.T.Charnock, T.A.Kennedy: “Combined optical and microwave approach for performing quantum spin operations on the nitrogen-vacancy in diamond”, *Physical Review B*, vol.64, Article number 041201, (2001)

[Chi00] A.M.Childs, I.L.Chuang: “Universal quantum computation with two-level trapped ions”, *Physical Review A.*, vol.63, 012306, (2000)

[Chu95] I.L.Chuang, Y.Yamamoto: “Simple quantum computer”, *Physical Review A*, vol.52, pp.3489, (1995)

[Chu98] I.L. Chuang, L.M.K. Vandersypen et al. : “Experimental realization of a quantum algorithm”, *Nature*, vol. 393, pp.143-146, (1998)

[Cir95] J.I.Cirac, P.Zoller: “Quantum computation with cold trapped ions”, *Physical Review Letters*, vol.74,(20),pp.4091-4094,(1995)

[Cla08] J.Clarke, F.K.Wilhelm: "Superconducting quantum bits", *Nature*, vol.453, pp.1031-1042, (2008)

[Cle98] R.Cleve, A.Ekert et al.: "Quantum algorithms revisited", *Proceedings of the Royal Society of London Series A - Mathematical, Physical and Engineering Sciences*, vol. 454, pp.339-354, (1998)

[Col98] D. Collins, K.W. Kim et al.: "Deutsch-Jozsa algorithm as a test for quantum computation", *Physical Review A*, vol.58, pp 1633-1636, (1998)

[Cor97] D. G. Cory, A. F. Fahmy et al.: "Ensemble quantum computing by NMR spectroscopy", *Proceedings of the National Academy of Sciences of the United States of America*, vol.94, pp.1634-1639, (1997)

[Das05] S.Das Sarma, R.de Sousa et al.:"Spin quantum computation in silicon nanostructures", *Solid State Communication*, vol.133, pp.737-746, (2005)

[Deu85] D. Deutsch: "Quantum theory, the Church-Turing principle and the universal quantum computer", *Proceedings of the Royal Society of London Series A- Mathematical, Physical and Engineering Sciences*, vol. 400, pp.97-117, (1985)

[Deu92] D. Deutsch, R. Jozsa: "Rapid solution of problems by quantum computation", ",*Proceedings of the Royal Society of London Series A-Mathematical, Physical and Engineering Sciences*, vol.439, pp.553-558, (1992)

[DiV00] D.P.DiVincenzo: "The physical implementation of quantum computation", *Fortschritte der Physik*, vol.48, pp.771-783, (2000)

[Elz04] J.M.Elzerman, R.Hanson et al.: "Single-shot read-out of an individual electron spin in a quantum dot", *Nature*, vol.430, p.431-435, (2004)

[Fah08] A.F.Fahmy, R.Marx, W.Bermel, S.J.Glaser, "Thermal equilibrium as an initial state for quantum computation by NMR", *Physical Review A*, vol.78, Aug. 2008, Article number 022317

[Ger05] E.Gerjuoy: "Shor's factoring algorithm and modern cryptography. An illustration of the capabilities inherent in quantum computers", *American Journal of Physics*, vol.73, pp.521-540, (2005)

[Ger97] N. Gerschenfeld, I. L. Chuang: "Bulk spin-resonance quantum computation", *Science*, vol.275, pp.350-356, (1997) [Gol02] V.N.Golovach, D.Loss: “Electron spins in artificial atoms and molecules for quantum computation”, *Semiconductor Science and Technology*, vol.17, p.355, (2002)

[Gop08] T.Gopinath, A.Kumar, “Implementation of controlled phase shift gates and Collins version of the Deutsch-Jozsa algorithm on a quadrupolar spin-7/2 nucleus using non-adiabatic geometric phases”, *Journal of Magnetic Resonance*, vol.193, Aug.2008, pp.168-176

[Gul03] S.Gulde, M.Riebe et al.: “Implementation of the Deutsch-Jozsa algorithm on an ion trap quantum computer”, *Nature*, vol.421, pp.48-50,(2003)

[Haf08] H.Haffner, C.F.Roos, R.Blatt: “Quantum computing with trapped ions”, *Physics Reports-Review Section of Physics Letters*, vol.469, pp.155-203, (2008)

[Han08] R. Hanson, D.D.Awschalom: “Coherent manipulations of single spins in semiconductors”, *Nature*, vol.453, pp.1043-1049, (2008)

[Han09] R.Hanson: “Quantum information Mother Nature outgrown”, *Nature Materials*, vol.8, pp.368-369, (2009)

[Hog03] J.Hogan: “Quantum bits and silicon chips”, *Nature*, vol.424, pp.484-486, (2003)

[Hug98] R.J.Hughes, D.F.V.James, J.J.Gomez, M.S.Gulley, M.H.Holzscheiter, P.G.Kwiat et al.: “The Los Alamos trapped ion quantum computer experiment”, *Fortschritte der Physik*, vol.46, pp.329-361, (1998)

[Ito05] K.M.Itoh: “An all-silicon linear chain NMR quantum computer”, *Solid State Communications*, vol.133, pp.747-752, (2005)

[Jel02] F.Jelezko, I.Popa et al.: “Single spin states in a defect center resolved by optical spectroscopy”, *Applied Physics Letters*, vol.81, pp.2160-2162, (2002)

[Jel04a] F.Jelezko, T.Gaebel et al.: “Observation of coherent oscillations in single electron spin”, *Physical Review Letters*, vol.92, Article number 076401, (2004)

[Jel04b] F.Jelezko, T.Gaebel et al.: “Observation of coherent oscillation of a single nuclear spin and realization of a two-qubit conditional quantum gate”, *Physical Review Letters*, vol.93, Article number 130501, (2004)

[Jes04] P.S. Jessen, I.H.Deutsch et al.: “Quantum Information Processing with

Trapped Neutral Atoms", quant-ph/0404055, (2004)

[Jon98] J. A. Jones, M. Mosca: "Implementation of a quantum algorithm on a nuclear magnetic resonance quantum computer", *Journal of Chemical Physics*, vol.109, pp.1648-1653, (1998)

[Kan98] B.E. Kane: "A silicon-based quantum computer", *Nature*, vol.393, pp.133-137,(1998)

[Kie02] D.Kielpinski, C.Monroe et al.: "Architecture for a large-scale ion-trap quantum computer", *Nature*, vol.417, pp.709-711, (2002)

[Kim00] J. Kim, J.S. Lee et al.: "Implementation of the refined Deutsch-Jozsa algorithm on a three-bit quantum computer", *Physical Review A*, vol.62, 022312, (2000)

[Kop06] F.H.L.Koppens, C.Buizert et al.: "Driven coherent oscillations of a single electron in a quantum dot", *Nature*, vol.442, p.766-771, (2006)

[Kou06] L.P.Kouwenhoven, J.M. Elzerman et al.: "Control and measurement of electron spins in semiconductor quantum dots", *Physica Status Solidi B-Basic Solid State Physics*, vol.243, pp.3682-3691, (2006)

[Kou98] L.Kouwenhoven, C.Marcus: "Quantum Dots", *Physics World*, vol.11, p.35-39, (1998)

[Lad02] T.D.Ladd, J.R.Goldman et al.: "All-silicon quantum computer", *Physical Review Letters*, vol.89, Article number 017901, (2002)

[Lei03] D.Leibfried, B.DeMarco et al.: "Experimental demonstration of a robust, high-fidelity geometric two ion-qubit phase gate", *Nature*, vol.422, pp.412-415, (2003)

[Lei05] D.Leibfried, E.Knill, S.Seidelin et al.: "Creation of a six-atom 'Schrödinger cat' state", *Nature*, vol.438, pp.639-642, (2005)

[Lly96] S.Llyoyd: "Universal quantum simulators", *Science*, vol.273, pp.1073-1078, (1996)

[Los98] D.Loss, D.P.DiVicenzo: "Quantum computation with quantum dots", *Physical Review A*, vol.57, p.120-126, (1998)

[Mak99] Y.Makhlin, G.Schön, A.Shnirman: "Josephson-junction qubits with controlled couplings", *Nature*, vol.398, pp.305-307, (1999)

[Mak01] Y.Makhlin, G.Schön et al.: "Quantum-state engineering with Josephson-junction devices", *Review of Modern Physics*, vol.73, pp.357-400,(2001) [Mon95] C.Monroe, D.M.Meekhof et al.:“Demonstration of a fundamental quantum logic gate”, *Physical Review Letters*, vol.75, pp.4714-4718, (1995)

[Mon09] T.Monz, K.Kim, W.Hansel et al.: “Realization of the quantum toffoli gate with trapped ions”, *Physical Review Letters*, vol.102, Article number 040501, (2009)

[Mor08] J.J.L.Morton, A.M.Tyryshkin, R.M.Brown, S.Shankar, B.W.Lovett et al.: “Solid-state quantum memory using the ^{31}P nuclear spin”, *Nature*, vol.445, pp.1058-1088, (2008)

[Nak99] Y.Nakamura, Yu.A.Pashkin et al.: “Coherent control of macroscopic quantum states in a single-Cooper-pair box”, *Nature*, vol.398, pp.786-788,(1999)

[Neg06] C.Negrevergne et al.: “Benchmarking quantum control methods on a 12-qubit system”, *Physical Review Letters*, vol.96, Article number 170501, (2006)

[Nie03] M.A. Nielsen, I.L.Chunag: “Quantum computation and quantum information”, Cambridge: Cambridge University Press, (2003)

[Niz05] A.P.Nizovtsev, S.Ya.Kilin et al.: “A quantum computer based on NV centers in diamond: optically detected nutations of single electron and nuclear spins”, *Optics and Spectroscopy*, vol.99, pp.248-260, (2005)

[Pet05] J.R.Petta, A.C.Johnson et al.: “Coherent manipulations of coupled electron spins in semiconductor quantum dots”, *Science*, vol.309, p.2180-2184, (2005)

[Pla07] J.H. Plantenberg, P.C. de Groot, C.J.P.M Harmans et al.: “Demonstration of controlled-NOT quantum gates on a pair of superconducting quantum bits”, *Nature*, vol.446, pp.836-839,(2007)

[Pra08] S. Prawer and A.D.Greentree: “Applied physics – Diamond for quantum computing”, *Science*, vol.320, pp.1601-1602, (2008)

[Pre08] D.Press, T.D.Ladd, B.Zhang, Y.Yamamoto: “Complete quantum control of a single spin using ultrafast optical pulses”, *Nature*, vol.456, pp.218-221, (2008)

[Sch03a] F. Schmidt-Kaler, H.Häffner et al.: “Realization of the Cirac-Zoller controlled-NOT quantum gate”, *Nature*, vol.422, pp.408-411, (2003) [Sch03b] S.R.Schofield,N.J.Curson, M.Y.Simmons, F.J.Rueß, T.Hallam, L.Oberbeck, R.G.Clark: "Atomically precise placement of single dopants in Si", *Physical Review Letters*, vol.91, Article number 136104, (2003)

[Sch04] D.Schrader, I.Dotsenko et al.: "A neutral atom quantum register", *Physical Review Letters*, vol.93, Article number 150501, (2004)

[Shn97] A.Shnirman, G. Schön et al.:"Quantum manipulations of small Josephson junctions", *Physical Review Letters*, vol.79, pp.2371-2374,(1997)

[Sho94] P.W. Shor: "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer", *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*, edited by S.Goldwasser, (IEEE Computer Society, Los Alamos, Ca), pp.124-134, (1994)

[Sør00] A.Sørensen, K.Mølmer: "Entanglement and quantum computation with ions in thermal motion", *Physical Review A*, vol. 62, 022311, (2000)

[Spi06] T.P.Spiller, W.J.Munro: "Towards a quantum information technology industry", *Journal of Physics: Condensed Matter*, vol.118, pp.V1-V10, (2006)

[Ste01] M. Steffen, L.M.K Vandersypen et al.:"Toward quantum computation: a five qubit quantum processor", *IEEE Micro*, pp.25-54, (2001)

[Sto03] A.M.Stoneham, A.J.Fisher et al.:"Optically driven silicon-based quantum gates with potential for high-temperature operation", *Journal of Physics: Condensed Matter*, vol.15, L447-L451, (2003)

[Sto09] A.M. Stoneham, A.H.Harker, G.W.Morely: "Could one make a diamond-based quantum computer", *Journal of Physics: Condensed Matter*, vol.21, Article number 364222,(2009)

[Tur95] Q.A.Turchette, C.J.Hood et al.: "Measurement of conditional phase shifts for quantum logic", *Physical Review Letters*, vol.75, pp.4710-4713, (1995)

[Tyr03] A.M.Tyryshkin, S.A.Lyon et al.: "Electron spin relaxation times of phosphorus donors in silicon", *Physical Review B*, vol.68, Article number 193207, (2003)

[Van00] L.M.K Vandersypen, M.Steffen et al.: "Experimental realization of an order-finding algorithm with an NMR quantum computer", *Physical*

Review Letters, vol.85, No.25, pp.5452-5455, (2000)

[Van01] L.M.K.Vandersypen, M.Steffen et al.: “Experimental realization of Shor’s quantum factoring algorithm using nuclear magnetic resonance”, *Nature*, vol.414, pp. 883-887, (2001)

[Van04] L.M.K.Vandersypen, I.L.Chuang: “NMR techniques for quantum control and computation”, *Review of Modern Physics*, vol.76, pp.1037-1069, (2004)

[Vri00] R.Vrijen, E.Yablonovitch et al.: “Electron-spin-resonance transistors for quantum computing in silicon-germanium heterostructures”, *Physical Review A*, vol.62, Article number 012306, (2006)

[Wei03] D.J.Weinland, M.Barrett et al.; “Quantum information processing with trapped ions”, *Philosophical Transactions of the Royal Society of London Series A-Mathematical, Physical and Engineering Sciences*, vol.361, pp.1349-1361, (2003)

[Wen07] G.Wendin, V.S.Shumeiko: “Quantum bits with Josephson Junctions”, *Low Temperature Physics*, vol.33, pp.724-744, (2007)

[Wra01] J.Wrachtrup, S.Ya.Kilin, A.P.Nizovtsev: “Quantum computation using the ^{13}C nuclear spins near the single NV defect center in diamond”, *Optics and Spectroscopy*, vol.91, pp.429-437, (2001)

[Wra06] J.Wrachtrup, F.Jelezko: “Processing quantum information in diamond”, *J. Phys.: Condens. Matter*, vol.18, S807-S824, (2006)

[Yam03] T.Yamamoto, Yu.A.Pashkin et al.: “Demonstration of conditional gate operation using superconducting charge qubits”, *Nature*, vol.425, pp.941-944, (2003)

[Zol05] P.Zoller, T.Beth et al.: “Quantum information processing and communication - Strategic report on current status, visions and goals for research in Europe”, *European Physical Journal D*, vol.36, pp.203-228, (2005)

Chapter 4 The SFG quantum logic gate

As seen in the previous chapter, although a number of different implementations of quantum computational systems have been proposed (and even tested experimentally), large-scale quantum computation is still far from the implementation stage and it is not yet clear which physical system might prove best[Zol05]. Further, while solid-state implementations are believed to have the potential for achieving a high degree of scalability due to the knowledge and expertise acquired through classical electronics, a problem common to some of these proposals seems to lie in the implementation of interactions between the qubits in a quantum register. In this chapter, the SFG model[Sto03] is described which proposes to solve this problem mediating the interaction between adjacent qubits through the optical excitation of a control particle placed in their proximity. This proposal aims at achieving scalability by exploiting the promising decoherence times of donor electron spin qubits in semiconductors[Tyr03] in conjunction with the optical control, which allows one to avoid control electrodes placed on the computation chip (used, for example, in Kane's proposal[Kan98]) with its high-precision fabrication techniques[Sto03]. This model may be able to control patches of about 20 qubits while larger quantum registers could be built by connecting different patches together[Sto08]. The remainder of this thesis focuses on the design of quantum logic circuits exploiting two-qubit SFG quantum logic gates.

4.1 The SFG model

In a quantum computer based on SFG quantum logic gates, the qubits are carried by the spin of an electron from a donor in a semiconductor substrate located in a magnetic field. The latter defines the reference direction for the qubits' spin alignment. The interactions between two qubits are mediated by a control particle positioned in their proximity. The control electron starts in a known state, for example, spin-up or spin-down. In their ground states, the wavefunctions of this three-particle system are separated and no interaction between them occurs. If, however, the electron of the control particle is brought to an excited state through an optical pulse, its wavefunction overlaps with the wavefunctions of the qubits, leading to an effective interaction between them. Following [Sto03,Rod04], assuming to have two qubits *A* and *B* and a control particle *C*, the interaction between the qubits and the control particle in its excited state can be modelled as an effective Heisenberg interaction through the Hamiltonian:

$$H_{Ham} = J_A \sigma_A \cdot \sigma_C + J_B \sigma_B \cdot \sigma_C + B_A \sigma_{Az} + B_B \sigma_{Bz} + B_C \sigma_{Cz} \qquad (4.1)$$

where J_A and J_B describe, respectively, the strength of the exchange interaction between qubit *A* and the control particle and qubit *B* and the control particle, B_i=-μ_iB with μ_i (i=*A*,*B*,*C*) the magnetic moment for the three particles and B the static magnetic field on the computation register. The σ_i terms are vectors of the Pauli *X*,*Y* and *Z* matrices for the three particles and σ_{iz} is the Pauli *Z* matrix.

The interaction between the qubits and the control particle is terminated by a second (de-exciting) pulse, which returns the electron of the control atom to its ground state after a time *T*. In order to have negligible interaction between the qubits and the control when the latter is in its ground state and sufficient interaction between the three particles when the control is brought to the excited state, distances between the control and the qubits are expected to be in the order of 10-20nm[Rod04,Sto08].

As described in [Ker07], the unitary transformation *U* which describes the impact on the qubits and control particle of the Hamiltonian shown in (4.1) during the pulse-interleave time *T* between exciting and de-exciting pulse can be described, in general terms, by the expression given in (4.2). This expression has been obtained assuming the state $|0\rangle$ to be associated with the spin-up state of a particle and $|1\rangle$ with spin-down. Further, the matrix is built assuming the particles to be ordered as $|C\, Q_A\, Q_B\rangle$ and the corresponding states sorted in ascending order. This choices lead to the following representation of the unitary matrix *U* of this transformation:

$$U = e^{-iH_{Ham}T} = \begin{bmatrix} U_+ & C_{M1} \\ C_{M2} & U_- \end{bmatrix} \qquad (4.2)$$

U_+ and U_- describe two 4x4 matrices which operate on the qubits' states, without leading to changes in the state of the controls. U_+ refers to the control electron being spin-up, while U_- to it being spin-down. Conversely, the two matrices C_{M1} and C_{M2} are 4x4 matrices which lead to an interaction between the states of the qubits and of the controls. In order to avoid entanglement of the control particle with the qubits and consequent loss of quantum information from the qubits to the controls, solutions for C_{M1} and C_{M2} with all coefficients equal to 0 are envisaged for the SFG protocol. As described in [Sto03], of all possible pulse-interleave times T between exciting and de-exciting pulses, only a discrete set defined by two integers M and N, produces entangling gates which satisfy this condition. Further, these solutions also return the control particle back to the state it was prior to the beginning of the interaction. As demonstrated in [Rod04], assuming the qubits to be symmetrically distributed with respect to the control particle, the transformation produced by the SFG protocol is then described by following expressions:

$$U_+(M,N)=e^{i(J-B)T}\begin{bmatrix} e^{-i[(3-f)J+2B]T} & 0 & 0 & 0 \\ 0 & \frac{\left[(-1)^M+e^{-i(1-f)JT}\right]}{2} & \frac{\left[(-1)^M-e^{-i(1-f)JT}\right]}{2} & 0 \\ 0 & \frac{\left[(-1)^M-e^{-i(1-f)JT}\right]}{2} & \frac{\left[(-1)^M+e^{-i(1-f)JT}\right]}{2} & 0 \\ 0 & 0 & 0 & e^{2iBT}(-1)^N \end{bmatrix} \tag{4.3a}$$

$$U_-(M,N)=e^{i(J+B)T}\begin{bmatrix} e^{-2iBT}(-1)^M & 0 & 0 & 0 \\ 0 & \frac{\left[(-1)^N+e^{-i(1+f)JT}\right]}{2} & \frac{\left[(-1)^N-e^{-i(1+f)JT}\right]}{2} & 0 \\ 0 & \frac{\left[(-1)^N-e^{-i(1+f)JT}\right]}{2} & \frac{\left[(-1)^N+e^{-i(1+f)JT}\right]}{2} & 0 \\ 0 & 0 & 0 & e^{-i[(3+f)J-2B]T} \end{bmatrix} \tag{4.3b}$$

with:

$$\begin{aligned} f&=\frac{B}{J}=-\frac{M^2+N^2}{M^2-N^2}\pm\sqrt{\left(\frac{M^2+N^2}{M^2-N^2}\right)^2-9} \\ JT&=\frac{M\pi}{\sqrt{(f-1)^2+8}}=\frac{N\pi}{\sqrt{(f+1)^2+8}} \\ BT&=\frac{M\pi}{\sqrt{\left(1-\frac{1}{f}\right)^2+\frac{8}{f^2}}}=\frac{N\pi}{\sqrt{\left(1+\frac{1}{f}\right)^2+\frac{8}{f^2}}} \end{aligned} \tag{4.4}$$

where, for the symmetric case, $J=J_A=J_B$ and $B=B_A=B_B$.

Again, $U_+(M,N)$ operates on the qubits when the control electron starts in the spin-up state, while the transformation $U_-(M,N)$ is applied if the control electron is in the spin-down state. As will be discussed in more detail in Section 4.2, from expressions (4.3a) it can be understood that different two-qubit gates can be implemented with the SFG scheme by changing the parameters given in expressions (4.4).

For example, in [Rod04] it was shown that, assuming the control atom to start in the spin-up state, for M=1584 and N=2177 one obtains f=4.5, JT=1105.84 and expression (4.3a) returns the controlled-phase (or controlled-Z) gate[Nie03]:

$$CP = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix} \qquad (4.5)$$

In reality, the controlled-phase gate obtained with M=1584 and N=2177 is not ideal, but an accurate approximation with an error of the order of 10^{-6}[Rod04], thus showing that SFG two-qubit gates can approximate gates typically used in literature with high precision.

When considering a quantum register of many qubits, the SFG model exploits randomness in the spatial distribution of the particles in order to be able to selectively address a specific control particle with its corresponding qubits. The main contribution to the value of the excitation frequency of a control particle will come from the choice of the host material and the dopants. Specifically, the excitation frequency will depend on the energy levels of the control particle in a given substrate. However, in a random distribution, particles will have different environments leading to slightly different energy levels and guaranteeing thereby the individual selectivity of each two-qubit gate [Sto03]. The selection of an individual control atom-qubit system (and, therefore, of a specific quantum gate) is shown schematically in Figure 4-1 for a 3-qubit quantum register.

In terms of signal parameters, in order to be sure that the excited control electron correctly couples with all information-carrying electron-spin states of the qubits, the excitation pulses need to have a sufficiently broad spectrum which typically translates into pulsewidths of the order of picoseconds[Rod04]. As will be discussed at the end of this section and in Chapter 6, values of the excitation frequencies of systems currently under study range from the far-infrared to the mid-infrared part of the spectrum.

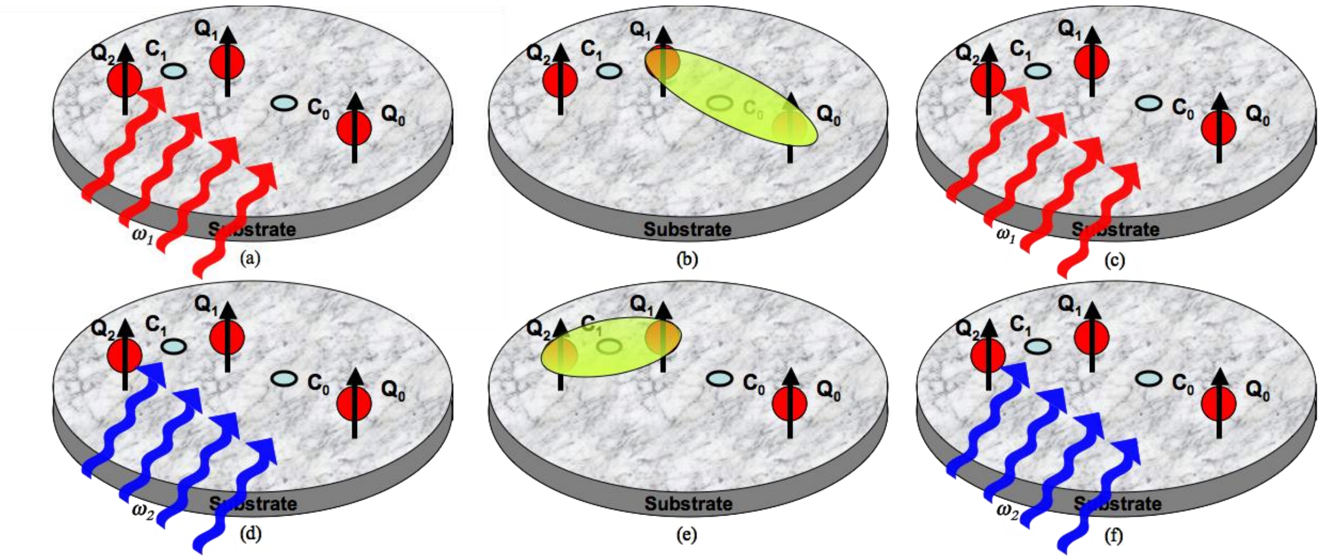


Figure 4-1: Schematic of the excitation of two SFG gates during computation. (a) An optical pulse centred on frequency ω_1 is incident on the computation chip. (b) The optical pulse excites control atom C$_0$ and triggers the interaction between qubits Q$_0$ and Q$_1$. (c)A second optical pulse centred on ω_1 de-excites the control atom, interrupting the interaction of the qubits. (d)To activate the second SFG gate, a pulse centred on ω_2 is transmitted on the chip.(e)The optical pulse excites control atom C$_1$ triggering the interaction between Q$_1$ and Q$_2$. (f)A second pulse centred on ω_2 de-excites C$_1$ terminating the interaction of the qubits.

Summarising, in the SFG scheme interactions between two qubits are introduced by transmitting pairs of optical pulses centred on specific frequencies ω_i and spaced by pulse-interleave times T. Further, single-qubit operations can be implemented, as in the other implementations based on electron spin qubits seen in Chapter 3, by transmitting microwave pulses centred on the resonance frequencies of the electron spin qubits. Hence, experimentally, a quantum computation exploiting the SFG model corresponds to the transmission on the quantum register of a well-defined sequence of multi-wavelength, picosecond optical pulses for controlling the two-qubit interactions and of microwave pulses for single-qubit operations.

Read-out, too, could be performed optically, with a technique similar to the one used in ion-trap computation, and reviewed in the previous chapter. The read-out protocol requires an auxiliary qubit and control particle which will be referred to, respectively, as Q_{-1} and C_{-10}, both starting in a known state, for example, $|0\rangle$ (spin-up), as shown schematically in Figure 4-2 for a three-qubit example.

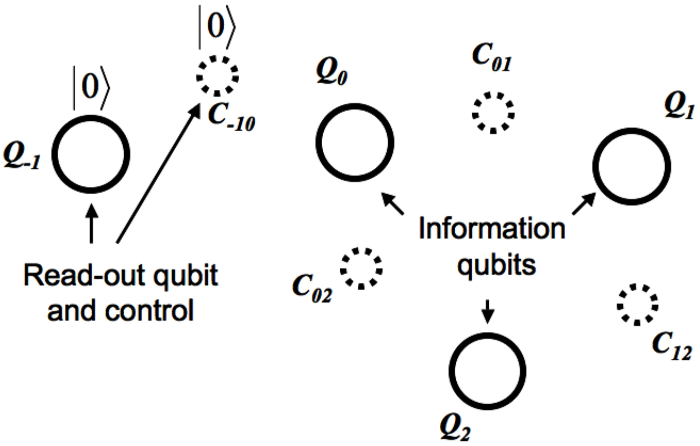


Figure 4-2: Auxiliary qubit and control particle used for read-out in a three-qubit quantum register example

The value of qubit Q_0 can be determined by analysing its state in conjunction with qubit Q_{-1} and control C_{-10}. At the beginning of the read-out procedure this sub-system can be described as being in the state $|C_{-10}Q_{-1}Q_0\rangle_g = |00Q_0\rangle_g$ where the subscript "*g*" refers to the control electron being in its ground state. If a narrow-band optical pulse, tuned to the transition $|000\rangle_g \leftrightarrow |000\rangle_e$ (where *"e"* refers to the control electron being in the excited state) is transmitted on the chip, excitation of the control electron to the state $|000\rangle_e$ will occur only if the qubit Q_0 is in the state $|0\rangle$. In this case, a long pulse will provide repeated excitation and de-excitation of the control electron with a consequent release of a photon after each de-excitation event. These scattered photons can be collected by photodetectors. If, however, Q_0 is in $|1\rangle$, the three-particle system is not resonant with the $|000\rangle_g \leftrightarrow |000\rangle_e$ transition, no photons would be emitted and, therefore, no signal would be produced by the photodetectors. Hence, the state of Q_0 can be determined by tuning the emission frequency of a laser to the $|000\rangle_g \leftrightarrow |000\rangle_e$ transition and by understanding whether or not the control electron couples to this transition by measuring the presence of scattered photons[Sto03]. Once the state of Q_0 is known (and remembering that in the SFG protocol the state of the control particles is known), the same concept can be applied using Q_0 and C_{01} for determining the state of Q_1. Hence, one after the other, the state of all the qubits in the quantum register can be identified.

To date, different materials have been investigated for building a quantum computer based on SFG gates[Sto08]. Key factors are the decoherence times of, both, the qubits and the control particles, as well as the ease with which the optical frequencies needed for the excitation can be generated. One possibility, which is currently under study, is the use of phosphorus as control-particles in a silicon substrate. However, while experiments on this system have generated valuable information on the experimental gate dynamics[Vin08], its excitation frequencies are in a far-infrared range (~36μm) in which compact and integrated laser sources have not been developed yet. Other possibilities are being considered (the double donor selenium in silicon or phosphorus particles in diamond, for example,[Sto08]) for which excitation frequencies fall in the mid-infrared part of the spectrum (~2.2-2.3μm, [Ber89,Laz08]) where, as will be discussed in Chapter 6, more convenient laser sources than for the case of phosphorus control particles in silicon are available.

4.2 Entangling characteristics of the SFG gate

From the equations given in expressions (4.3a) it can be seen that different entangling gates can be produced by the SFG model by changing the parameters in (4.4). Since the aim of the work presented in this thesis was to develop quantum logic circuits suitable for the experimental implementation of a quantum computation prototype based on SFG technology, it was important to study which quantum gates can be produced within this model in order to know what operations would have been available when designing circuits based on this model. This has been done here applying the geometrical method proposed by Zhang et al.[Zha03] (which is based on the study of the entangling characteristics of two-qubit gates through their visualization in a 3-dimensional space) to SFG gates. This method is based on the observation, demonstrated in [Zha03] by the authors, that any two-qubit gate U can be described mathematically by the expression:

$$U = k_1 e^{c_1 X_1 X_2 + c_2 Y_1 Y_2 + c_3 Z_1 Z_2} k_2 \qquad (4.6)$$

where k_1 and k_2 are operators which only act on single qubits, and, therefore, do not influence the entangling characteristics of the gate, while all the information on the entanglement power of the operator is stored in the exponential containing the three parameters c_1, c_2 and c_3 and the Pauli matrices X_i, Y_i and Z_i. The subscript in each Pauli matrix refers to which of the two qubits the matrix is applied. Given a number of different gates, by calculating the c_1, c_2 and c_3 parameters for each gate and by identifying the corresponding point (c_1, c_2, c_3) in a 3-dimenasional space which contains all the entangling gates (called the a^+ Weyl chamber[Zha03]), this method allows one to determine graphically, for example:

-Whether two apparently different two-qubit gates (i.e. different in terms of the unitary matrix U which describes them) actually implement an equivalent two-qubit gate (i.e. characterised by the same c_1, c_2 and c_3 parameters).

-Whether or not a two-qubit gate actually introduces any entanglement between two qubits.

-Whether a given two-qubit gate is a perfect entangler (i.e. a single application of the gate can bring an unentangled state of the qubits into a maximally entangled one[Zha03]).

These coefficients can be evaluated with the help of the G_1 and G_2 parameters defined by Makhlin in [Mak02] which, similarly to c_1, c_2 and c_3, have been derived for assessing the entangling characteristics of two-qubit gates. Given a matrix M describing a two-qubit operation, the G_1 and G_2 parameters for matrix M can be calculated by, first, expressing M in the Bell basis through the transformation $M \rightarrow M_B = Q^{\dagger} M Q$ with:

$$Q = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 & 0 & i \\ 0 & i & 1 & 0 \\ 0 & i & -1 & 0 \\ 1 & 0 & 0 & -i \end{bmatrix} \qquad (4.7)$$

and then by evaluating the matrix $m = M_B^{\dagger} M_B$. As demonstrated by Makhlin, the eignevalues of m are not changed by single-qubit operations applied to M and can therefore be used to analyse the entanglement produced by a given two-qubit transformation. Further, the spectrum of m is completely characterised by the two parameters:

$$\begin{aligned} G_1 &= \frac{(tr(m))^2}{16 \det M} \\ G_2 &= \frac{(tr(m))^2 - tr(m^2)}{4 \det M} \end{aligned} \qquad (4.8)$$

where *tr* is the trace operator of a matrix det its determinant.

Hence, G_1 and G_2 well describe the entangling characteristics of a two-qubit gate M.

As described by Zhang et al. in [Zha03], the c_1, c_2 and c_3 parameters are related to G_1 and G_2 by the expressions:

$$\begin{aligned}
G_1 &= \cos^2(c_1)\cos^2(c_2)\cos^2(c_3) - \sin^2(c_1)\sin^2(c_2)\sin^2(c_3) \\
&\qquad + \frac{i}{4}\sin(2c_1)\sin(2c_2)\sin(2c_3) \\
G_2 &= 4\cos^2(c_1)\cos^2(c_2)\cos^2(c_3) - 4\sin^2(c_1)\sin^2(c_2)\sin^2(c_3) \\
&\qquad - \cos(2c_1)\cos(2c_2)\cos(2c_3)
\end{aligned} \tag{4.9}$$

The system given in (4.9) can be solved for the case of SFG gates by taking the expressions of G_1 and G_2 for the SFG model (which have been derived in [Rod04] by evaluating equations (4.8) for the SFG two-qubit transformation shown in (4.3a), i.e. assuming the control electron to start in the spin-up state in a computation):

$$\begin{aligned}
G_1(M,N) &= \frac{(-1)^{(M+N)}\left[e^{-JT} + (-1)^N e^{iJT}\cos(1-f)JT\right]^2}{4} \\
G_2(M,N) &= (-1)^{(M+N)}\left[\cos(2JT) + 2(-1)^N\cos(1-f)JT\right]
\end{aligned} \tag{4.10}$$

and noticing that expressions (4.9) and (4.10) and can be respectively rearranged as:

$$\begin{aligned}
&4\Re(G_1) - G_2 = \cos(2c_1)\cos(2c_2)\cos(2c_3) \\
&4\Im(G_1) = \sin(2c_1)\sin(2c_2)\sin(2c_3) \\
&G_2 = \cos(2c_1) + \cos(2c_2) + \cos(2c_3)
\end{aligned} \tag{4.11}$$

and

$$\begin{aligned}
&4\Re\big(G_1(M,N)\big) - G_2(M,N) = (-1)^{(M+N)}\cos(2JT)\cos^2\big(JT(1-f)\big) \\
&4\Im\big(G_1(M,N)\big) = -(-1)^{(M+N)}\sin(2JT)\sin^2\big(JT(1-f)\big) \\
&G_2(M,N) = (-1)^{(M+N)}\cos(2JT) + 2(-1)^M\cos^2\big(JT(1-f)\big)
\end{aligned} \tag{4.12}$$

The c_1, c_2 and c_3 parameters for SFG gates can then be evaluated by inspection of equations (4.11)-(4.12) obtaining:

$$\begin{aligned}
c_1 &= \pi + (M+N)\frac{\pi}{2} - JT \\
c_2 &= M\frac{\pi}{2} - \frac{1}{2}JT(1-f) \\
c_3 &= M\frac{\pi}{2} - \frac{1}{2}JT(1-f)
\end{aligned} \tag{4.13}$$

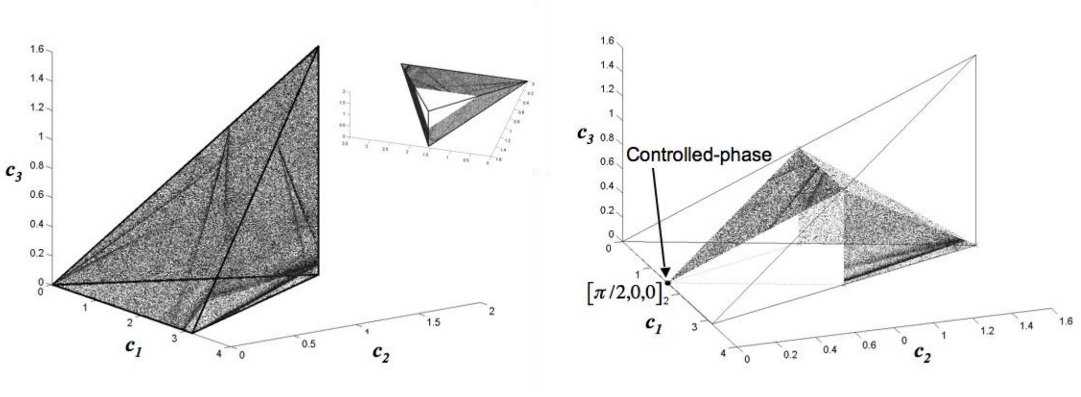
To plot the c_1, c_2 and c_3 parameters given above in the a^+ chamber other considerations must be taken into account. The characteristic of the a^+ chamber is that each point represents an ensemble of gates said to be *locally equivalent*. Two gates U_1 and U_2 are said to be locally equivalent if there exist some single-qubit operations $A \otimes B$ and

$C\otimes D$ such that $U_1 = A\otimes B\cdot U_2\cdot C\otimes D$ (i.e. the two transformations are equivalent up to some single-qubit operations). Given two locally equivalent SFG gates, the equations in expression (4.13) may, nevertheless, return different values of c_1, c_2 and c_3. Hence, the triplets returned by expression (4.13) need to be further reduced such that locally equivalent gates return the same c_i values. Also, to be inside the a^+ chamber, the triplets [c_1,c_2,c_3] have to satisfy the conditions:

$$\begin{aligned} &0 \le c_3 \le c_2 \le c_1 \\ &c_1 + c_2 \le \boldsymbol{\pi} \end{aligned} \tag{4.14}$$

As described in [Zha03], any triplet [c_1, c_2, c_3] can be reduced to the a^+ chamber observing that taking each c_i coefficient modulo π, permuting the three coefficients or applying any transformation of the type [c_i, π–c_j ,π– c_k], will not change the local equivalence of a triplet corresponding to a two-qubit gate. Hence, given a [c_1, c_2, c_3] triplet obtained, for example, through the equations given in (4.13), the corresponding point in the a^+ chamber can be found by, first, taking the triplet modulo π and by then trying any possible permutation or transformation of the type described above until the conditions given by expressions (4.14) are met.

Figure 4-3(a) shows the a^+ chamber and the c_1, c_2 and c_3 points evaluated for SFG gates with M and N values between 1 and 500. Each point represents a different entangling gate. As can also be seen from the inset in Figure 4-3(a), in which the upper points have been removed in order to look inside the chamber, the points cover the entire surface of the a^+ chamber (with the exception of the bottom face of this tetrahedron), showing that the SFG model can implement a variety of different entangling gates.



(a) (b)

Figure 4-3:(a) c_1, c_2 and c_3 coefficients in the a^+ chamber for SFG gates having M and N between 1 and 500. (b) Perfectly entangling SFG gates in the a+ chamber.

Further, the method proposed by Zhang et al. also allows one to visualise perfect entanglers, which, as mentioned above, are operators able to produce a maximally entangled state from a non-entangled one. In the a^+ chamber, perfect entanglers are represented by operators having c_i coefficients which satisfy one of the following two conditions[Zha03]:

$$\begin{aligned}&\frac{\pi}{2} \le c_i + c_k \le c_i + c_j + \frac{\pi}{2} \le \pi \\ &\frac{3\pi}{2} \le c_i + c_k \le c_i + c_j + \frac{\pi}{2} \le 2\pi\end{aligned} \tag{4.15}$$

where *i,j,k* are permutations of 1,2,3. The space defined by these equations and the corresponding SFG gates are shown in Figure 4-3(b). The ratio of the total number of entangling gates evaluated for large sets of *M* and *N* with the total number of perfect entanglers produced out of this set tends to 0.25. This is also the ratio of the total area uniformly covered by the SFG distribution with the area corresponding to perfectly entangling SFG gates, thus showing that about ¼ of the two-qubit gates which can be produced within the SFG model are perfect entanglers.

In the context of designing quantum circuits based on SFG gates these results are important for two reasons:

1) Firstly, they show that it should be possible to find convenient solutions of SFG gates approximating standard gates typically used in the literature. For example, the controlled-phase gate (or the C-NOT gate, since the C-NOT and the controlled-phase gate are locally equivalent[Zha03]) is characterised by the c_i triplet [$\pi/2, 0, 0$] which satisfies expressions (4.15) for perfect entanglement. Its location in the a^+ chamber is shown in Figure 4-3(b). From the relatively dense distribution of SFG gates on the surface of the a^+ chamber, it can be expected that a number of solutions close to the point corresponding to the controlled-phase gate will be available. These solutions represent different approximations of the controlled-phase gate which will differ in the degree of precision of the approximation and in the SFG gate parameters. This suggests that some flexibility in the choice of the SFG gate parameters will be available when approximating controlled-phase gates, which may allow one to balance, for example, precision with shortness of gate operation time, when choosing SFG gates for designing a quantum circuit.

2) At the same time, the fact that the whole surface of the a^{+} chamber is covered by points describing SFG gates suggests that there might be other entangling gates produced within this model, which may be convenient for designing circuits, for example, because of shorter gate operation times than those which can be achieved through SFG gates approximating controlled-phase gates.

These two considerations have been used in the work presented in Chapter 6 for testing different sets of SFG gates in the design of quantum circuits in order to explore strategies for developing circuits characterised by short computational times. Specifically, two sets of SFG gates approximating controlled-phase gates have been tested, which differed in the precision of approximation and in their gate computation time. Also, alternative entangling gates have been analysed which had gate computation times shorter than any of the previously considered approximations of controlled-phase gates.

4.3 Summary

The main characteristics of a quantum computational system based on SFG gates have been reviewed. SFG gates are solid-state two-qubit quantum logic gates which exploit the optical excitation of control particles placed in proximity of the qubits for mediating their interaction. Using a geometrical visualisation method, the two-qubit gates which can be produced within the SFG model have been analysed.

It is now possible to address in more detail the problem of quantum circuit design with SFG gates. As reviewed in the first chapters, mathematically, quantum logic gates can be described through matrices. Hence, to analyse quantum circuits and their design it is convenient to develop numerical tools which facilitate the generation and manipulation of matrices corresponding to specific quantum transformations. The next chapter describes the details of a quantum logic simulator and also of an automated quantum circuit design tool based on genetic programming, developed in the course of this work.

4.4 References

[Ber89] K. Bergman, G. Grossmann, H.G. Grimmeiss, M. Stavole, R.E. McMurray, “Applicability of the deformation-potential approximation to deep donors in silicon”, *Physical Review B*,vol. 39, pp.1104-1119, (1989)

[Kan98] B.E. Kane: “A silicon-based quantum computer”, *Nature*, Vol.393, pp.133-137,(1998)

[Ker07] A. Kerridge, A.H. Harker, A.M. Stoneham, “Electron dynamics in quantum gate operation”,*Journal of Physics: Condensed Matter*, vol. 19, Article number 282201, (2007)

[Laz08] A.Lazea, V.Mortet, J.D’Haen, P. Geithner, J. Ristein, M.D’Olieslaeger, K. Haenen, “Growth of polycrystalline phosphorus-doped CVD diamond layers”, *Chemical Physics Letters*, vol. 454, pp.310-313, (2008)

[Mak02] Y.Makhlin: “Nonlocal properties of two-qubit gates and mixed states and optimisation of quantum computation”, arXiv:quant-ph/0002045, (2002)

[Nie03] M.A.Nielsen, I.L.Chuang: “Quantum Computation and Quantum Information”, Cambridge University Press, (2003)

[Rod04] R.Rodriquez, A.J.Fisher, P.T.Greenland, A.M.Stoneham: “Avoiding entaglement loss when two qubit gates are controlled by electronic excitation”, *Journal of Physics: Condensed Matter*, Vol.16, pp.2757-2772,(2004)

[Sto03] A.M.Stoneham, A.J.Fisher, P.T.Greenland: “Optically driven silicon-based quantum gates with potential for high-temperature operation”, *Journal of Physics: Condensed Matter*, Vol.15, L447-L451, (2003)

[Sto08] A.M.Stoneham: “The quantum in your materials world”, *Materials Today*, Vol.11, pp.32-36, (2008)

[Tyr03] A.M.Tyryshkin, S.A.Lyon et al.: “Electron spin relaxation times of phosphorus donors in silicon”, *Physical Review B*, vol.68, Article number 193207, (2003)

[Vin08] N.Q. Vinh, P.T.Greenland, K. Litvinenko, B. Redlich, A.F.G van der Meer, S.A. Lynch, et al.: “Silicon as a model ion trap: Time domain measurements of donor Rydberg states”, *Proceedings of the National Academy of Sciences of the United States of America*, vol.105, pp.10649-10653, (2008) [Zha03] J.Zhang, S.Sastry, K.B.Whaley: "Geometric theory of nonlocal two-qubit operatrions", *Physical Review A*, Vol.67, Article number 042313, (2003)

[Zol05] P.Zoller, T.Beth et al.: "Quantum information processing and communication - Strategic report on current status, visions and goals for research in Europe", *European Physical Journal D*, vol.36, pp.203-228, (2005)

Chapter 5 Numerical tools for the analysis and design of quantum logic circuits

This chapter describes two numerical tools which have been developed in the course of the work described in this thesis. The first one is a quantum logic simulator which analyses the performance of a quantum circuits. The simulator has been developed specifically for quantum circuits based on SFG gates, described in Chapter 4, as it not only analyses the evolution of qubits but also how control particles are affected by the computation. The second tool implements a quantum circuit design algorithm based on a genetic programming approach. Both tools have been realised in Matlab and have been used to obtain the results presented in the next chapter.

5.1 The quantum logic simulator

When focusing on the design of quantum logic circuits, it is important to have tools for assessing how well a given circuit implements a desired transformation, for comparing how different circuits implement a given function or to model how the performance of a circuit may decay in case of non-ideal behaviour of the gates comprised in the circuit. Algorithms and protocols for simulating the behaviour of quantum logic circuits have been described, for example, in [Obe99,Sch00]. Starting from a given input state of a set of qubits in a quantum register, quantum logic simulators typically model the evolution of the quantum logical states of the qubits under the influence of quantum logic gates, measurements for the read-out and the effects of errors introduced by the non-ideal control of the devices used during computation, which may lead to the imperfect implementation of quantum gates (operational errors), or decoherence. However, as described in the previous chapter and as will be described in more detail below, in SFG quantum computation, the transformations applied to qubits also depend on the state of the control particles. Hence, in order to obtain a quantum logic simulator which could be used in this project for analysing the performance of quantum circuits based on the SFG proposal, models such as the ones described in [Obe99,Sch00] have been modified so as to consider the specific dynamics of SFG gates and the evolution of the states of the control particles.

This resulted in the implementation of a numerical tool specifically designed for analysing the performance of circuits based on SFG gates and its structure was kept as flexible as possible in order to allow the gate libraries or error models to be easily updated in the course of the work and subsequently. In the previous chapter the main features and entangling characteristics of an SFG gate have been studied as a single unit. Instead, the quantum logic simulator was used to analyse the performance of SFG gates when part of a complete quantum circuit.

As described in Chapter 2, a quantum computation is typically performed by applying a well-defined sequence of single- and two–qubit gates, i.e. the quantum circuit, to a quantum register. Mathematically, starting from the initial state of the register, given through the probability amplitudes c_{in_i} of the input state vector $|\psi_{in}\rangle$:

$$|\;\psi_{in}\rangle = \begin{pmatrix} c_{in_1} \\ c_{in_2} \\ \cdot \\ \cdot \\ \cdot \\ c_{in_{2^n}} \end{pmatrix}; \qquad \text{normalised such that } \sum_1^{2^n} \left|c_{in_i}\right|^2 = 1 \qquad (5.1)$$

a computation can be studied by analysing how the state of the quantum register is affected by the unitary transformation U_{tot} corresponding to the quantum circuit. Further, the unitary transformation U_{tot} can be obtained by multiplying together all the unitary transformations corresponding to the sequence of single- and two-qubit gates implemented by the quantum circuit:

$$U_{tot} = U_m U_{m-1} U_{m-2} ......U_2 U_1 \qquad (5.2)$$

The output state $|\;\psi_{out}\rangle$ of the quantum register prior to any measurement can then be evaluated by multiplying the input state with the ordered sequence of unitary operators[Nie03]:

$$|\psi_{out}\rangle = \begin{pmatrix} c_{out_1} \\ c_{out_2} \\ \cdot \\ \cdot \\ \cdot \\ c_{out_{2^n}} \end{pmatrix} = U_{tot} \cdot |\psi_{\;in}\rangle = U_{tot} \begin{pmatrix} c_{in_1} \\ c_{in_2} \\ \cdot \\ \cdot \\ \cdot \\ c_{in_{2^n}} \end{pmatrix} = U_m U_{m-1} U_{m-2} ......U_2 U_1 \begin{pmatrix} c_{in_1} \\ c_{in_2} \\ \cdot \\ \cdot \\ \cdot \\ c_{in_{2^n}} \end{pmatrix} \qquad (5.3)$$

Table 5-1 and Table 5-2 show, respectively, the unitary matrices corresponding to some useful unitary transformations commonly used in many publications [Nie03, Sch03,Pet05,Jon98,Eke96,Bar95,Kim00a] and which have been included in the quantum logic simulator as part of the quantum gates library. The subscripts *x*,*y*,*z* refer to the Bloch sphere representation used in [Nie03] and reviewed in Chapter 2. *$R_x(\theta)$* describes, for example, a rotation of the qubit state by an angle *θ* about the *x*-axis of the Bloch-sphere.

Table 5-1: Single-qubit gates implemented in the quantum logic simulator

Name	**Matrix representation**
Rotation operators	$R_x(\theta)=\begin{bmatrix}\cos\left(\frac{\theta}{2}\right) & -i\sin\left(\frac{\theta}{2}\right)\\ -i\sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right)\end{bmatrix}$ $R_y(\theta)=\begin{bmatrix}\cos\left(\frac{\theta}{2}\right) & -\sin\left(\frac{\theta}{2}\right)\\ \sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right)\end{bmatrix}$ $R_z(\theta)=\begin{bmatrix}e^{-i\frac{\theta}{2}} & 0\\ 0 & e^{i\frac{\theta}{2}}\end{bmatrix}$
Hadamard gate	$H=\frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}$
Phase gate	$S=\begin{bmatrix}1 & 0\\ 0 & i\end{bmatrix}$
Pauli matrices	$X=\begin{bmatrix}0 & 1\\ 1 & 0\end{bmatrix}$ $Y=\begin{bmatrix}0 & -i\\ i & 0\end{bmatrix}$ $Z=\begin{bmatrix}1 & 0\\ 0 & -1\end{bmatrix}$
π/8	$T=e^{i\frac{\pi}{8}}\begin{bmatrix}e^{-i\frac{\pi}{8}} & 0\\ 0 & e^{i\frac{\pi}{8}}\end{bmatrix}=\begin{bmatrix}1 & 0\\ 0 & e^{i\frac{\pi}{4}}\end{bmatrix}$
Constant phase shift	$P(\vartheta)=e^{-i\vartheta}$

Table 5-2: Two-qubit gates implemented in the quantum logic simulator

Name	**Matrix representation**
Controlled-Not gate	$C-NOT=\begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{pmatrix}$
Controlled-Phase gate	$CP=\begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&-1\end{pmatrix}$
Root-swap gate	$\sqrt{SWAP}=\begin{pmatrix}1&0&0&0\\0&\frac{1-i}{2}&-\frac{1+i}{2}&0\\0&-\frac{1+i}{2}&\frac{1-i}{2}&0\\0&0&0&-1\end{pmatrix}$
SFG gate	Equations and details given in Chapter 4
J_{NMR} (two-qubit gate in nuclear magnetic resonance)[Nie03,Kim00a]	$J_{NMR}(\vartheta)=e^{-\frac{i}{2}\vartheta Z\otimes Z}$

5.1.1 Changing the states of the input register

As can be seen from Table 5-1 and Table 5-2 all given transformations can be expressed in terms of 2x2 and 4x4 matrices. However, even a simple single-qubit operation changes all the coefficients of the 2^n long state vector (e.g. $|\psi_{out}\rangle$ in equation (5.3)) used for describing the state of an n-qubit quantum register during a computation. The most straightforward way to compute an output state vector given an input state vector and a unitary operation on a subset of qubits from the quantum register is to build 2^nx2^n matrices using the Kronecker product $\otimes$, which, on two matrices A and B, returns[Nie03]:

$$A\otimes B=\begin{bmatrix}a_{11}&a_{12}\\a_{21}&a_{22}\end{bmatrix}\otimes\begin{bmatrix}b_{11}&b_{12}\\b_{21}&b_{22}\end{bmatrix}=\begin{bmatrix}a_{11}b_{11}&a_{11}b_{12}&a_{12}b_{11}&a_{12}b_{12}\\a_{11}b_{21}&a_{11}b_{22}&a_{12}b_{21}&a_{12}b_{22}\\a_{21}b_{11}&a_{21}b_{12}&a_{22}b_{11}&a_{22}b_{12}\\a_{21}b_{21}&a_{21}b_{22}&a_{22}b_{21}&a_{22}b_{22}\end{bmatrix} \quad (5.4)$$

For example, given a single-qubit operation (described by a 2x2 operator U_{2x2}) to be applied on the k^{th} qubit, the corresponding 2^nx2^n matrix can be built by multiplying left the U_{2x2} operator with *n-(k+1)* 2x2 identity matrices through the Kronecker product and multiplying it on the right side with *k* 2x2 identity matrices, again, via a Kronecker product:

$$U_{2^n x 2^n} = I_{n-1} \otimes I_{n-2} \otimes ....I_{k+1} \otimes U_{2x2} \otimes I_{k-1} \otimes .... \otimes I_1 \otimes I_0 \qquad (5.5)$$

Once the 2^nx2^n matrix has been computed through expression (5.5), the evolution of the quantum state vector can be computed by applying equation (5.3). The drawback of this method is that, even with single-qubit operations, it is still necessary to perform the whole multiplication of the corresponding 2^nx2^n matrix (expression (5.5)) with the 2^n state vector describing the register. A more efficient way, especially when dealing with unitary transformations on a small number of qubits compared to the size of the quantum register, is described in [Sch00]. The method is based on following idea:

Given a quantum register in the state:

$$\begin{aligned} &|\psi\rangle = c_0|00..00\rangle + c_1|00..01\rangle + ... \\ &\quad + c_i|a_{n-1}a_{n-2}..a_1a_0\rangle + ... + c_{2^n-1}|11..11\rangle \qquad (5.6) \\ &a_i \in \{0,1\} \end{aligned}$$

and

$$U_{2x2} = \begin{pmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{pmatrix} \qquad (5.7)$$

as the operator which has to be applied to the k^{th} qubit, using expressions (5.3)-(5.5), it is possible to see that, after having applied the $U_{2^n x 2^n}$ transformation corresponding to U_{2x2}, the probability amplitude:

$$c_i \qquad (5.8)$$

corresponding to the state

$$|a_{n-1}a_{n-2}..a_{k+1}0a_{k-1}..a_1a_0\rangle \qquad (5.9)$$

will be transformed into:

$$c_{i\,new} = u_{11} \cdot c_i + u_{12} \cdot c_{i+2^k} \qquad (5.10)$$

whereas the coefficient:

$$c_{i+2^k} \tag{5.11}$$

which corresponds to the state

$$\left|a_{n-1}a_{n-2}..a_{k+1}1a_{k-1}..a_1a_0\right\rangle \tag{5.12}$$

will be transformed into:

$$c_{i+2^k\,new} = u_{21}\cdot c_i + u_{22}\cdot c_{i+2^k} \tag{5.13}$$

This can be expressed in the compact form:

$$\begin{pmatrix} c_{i\,new} \\ c_{i+2^k\,new} \end{pmatrix} = U_{2x2}\begin{pmatrix} c_i \\ c_{i+2^k} \end{pmatrix} \tag{5.14}$$

It is therefore possible, instead of evaluating all coefficients c_i simultaneously using a 2^nx2^n matrix (i.e. expression (5.5) followed by expression (5.3)), to evaluate expression (5.14), for all the $2^n/2$ pairs of coefficients c_i and c_{i+2^k} which form the state vector. This method is implemented by evaluating expression (5.14) for c_0 and c_{0+2^k} and by copying the obtained values into, respectively, position 1 and 1+2^k of the new state vector. Then, evaluating expression (5.14) for c_1 and c_{1+2^k} and copying the new values in position 2 and 2+2^k of the new state vector, and so on until the whole state vector has been processed. While the multiplication of the complete state vector with the 2^nx2^n matrix shown in expression (5.5) has a complexity of $O(2^{2n+1})$, the algorithm described in [Sch00] reduces the complexity to $O(2^{n+1})$ and has therefore been implemented in the quantum logic simulator here presented.

This algorithm can also be generalized for the case of l-qubit quantum gates:

$$U_{2^l x 2^l} = \begin{pmatrix} u_{1,1} & \dots & \dots & \dots & u_{1,2^l} \\ \cdot & & & & \cdot \\ \cdot & & & & \cdot \\ \cdot & & & & \cdot \\ u_{2^l,1} & \dots & \dots & \dots & u_{2^l,2^l} \end{pmatrix} \tag{5.15}$$

Considering a set of l ordered indices $\alpha<\beta<...<\gamma$ describing the positions of the qubits in the quantum register to which the gate has to be applied, the algorithm here requires to extract from the state vector the coefficients c_i, c_{i+2^α} ….. $c_{i+2^\alpha+2^\beta}$, $c_{i+2^\alpha+2^\beta+...+2^\gamma}$ to perform the transformation:

$$\begin{pmatrix} c_{inew} \\ c_{i+2^{\alpha}new} \\ \cdots\cdots \\ c_{i+2^{\beta}+2^{\alpha}new} \\ \cdots\cdots \\ c_{i+2^{\gamma}+..+2^{\beta}+2^{\alpha}new} \end{pmatrix} = U_{2^l x 2^l} \begin{pmatrix} c_i \\ c_{i+2^{\alpha}} \\ \cdots\cdots \\ c_{i+2^{\beta}+2^{\alpha}} \\ \cdots\cdots \\ c_{i+2^{\gamma}+..+2^{\beta}+2^{\alpha}} \end{pmatrix} = \begin{pmatrix} u_{1,1} & \cdots & \cdots & \cdots & u_{1,2^l} \\ . & & & & . \\ . & & & & . \\ . & & & & . \\ u_{2^l,1} & \cdots & \cdots & \cdots & u_{2^l,2^l} \end{pmatrix} \begin{pmatrix} c_i \\ c_{i+2^{\alpha}} \\ \cdots\cdots \\ c_{i+2^{\beta}+2^{\alpha}} \\ \cdots\cdots \\ c_{i+2^{\gamma}+..+2^{\beta}+2^{\alpha}} \end{pmatrix} \qquad (5.16)$$

As in the single qubit case, the new values are then copied in the updated state vector and the algorithm proceeds with the set of coefficients corresponding to c_{i+1}, continuing until the whole state vector has been processed. In this case, the complexity is $O(2^{n-l}2^{2l+1})$ and has been implemented in the quantum logic simulator for the case of two-qubit gates.

To analyse a specific quantum circuit, its corresponding ordered gate sequence is given as input to the quantum logic simulator in the form of a text file. Each quantum gate of Table 5-1 and Table 5-2 is associated with a gate identifier. The structure of the simulation algorithm is as follows:

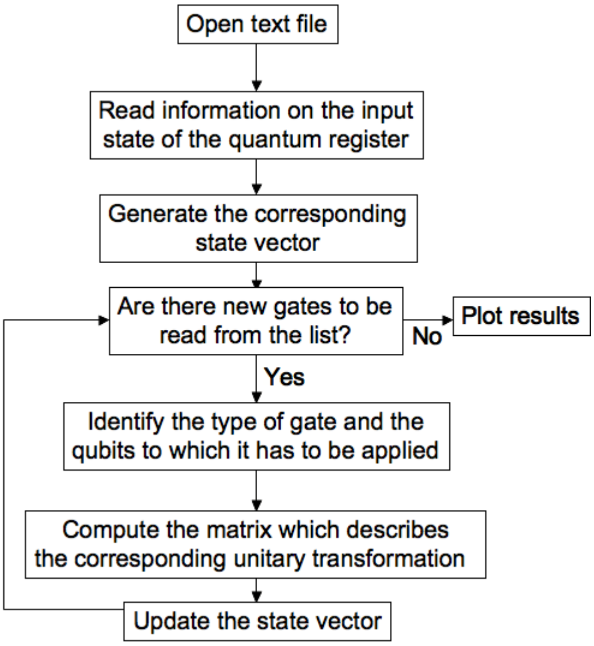

Open text file
Read information on the input state of the quantum register
Generate the corresponding state vector
Are there new gates to be read from the list?
No
Plot results
Yes
Identify the type of gate and the qubits to which it has to be applied
Compute the matrix which describes the corresponding unitary transformation
Update the state vector

Figure 5-1: Structure of the simulator

5.1.2 Simulating a quantum circuit based on SFG gates

As described in the previous chapter, in an SFG gate the interaction between two qubits *A* and *B* is mediated by a control atom *C*. It is started by the excitation of an electron from the control atom through an optical pulse and stops after the transmission of a de-exciting pulse after a pulse-interleave time *T*. The impact of this transformation can be described by the matrix:

$$U = e^{-iH_e T} = \begin{bmatrix} U_+ & C_{M1} \\ C_{M2} & U_- \end{bmatrix} \qquad (5.17)$$

where H_e is the Hamiltonian of the three-particle interaction in the excited state, U_+ and U_- are matrices which only change the state of the qubits, while C_{M1} and C_{M2} also change the state of the controls. In order to avoid entanglement between the qubits and controls at the end of the gate protocol, solutions with $C_{M1}=C_{M2}=0$ are exploited which, as demonstrated in [Rod04], depend on two integers *M* and *N* and can be summarised as:

$$U = \begin{bmatrix} U_+(M,N) & 0 \\ 0 & U_-(M,N) \end{bmatrix} \qquad (5.18)$$

Equation (5.18) can be interpreted as follows. The excitation of the control atom is equivalent to a unitary transformation being applied to the qubits. The nature of this transformation depends on the spin state of the control electron. If the control electron is in the spin-up state then U_+ is the transformation which will affect the qubits, otherwise it will be U_-. In an ideal case each qubit-control atom-qubit system will start with the control electron in a well defined state, spin-up or spin-down, so that it is known which of the transformations will be implemented, and will return after the excitation back to its starting state. It is possible to assume, therefore, that only U_+ or U_- will be applied to the quantum register. However, as will be discussed below, there might be situations in which the imperfect control over the system may lead to perturbations in the SFG gate protocol such that, instead of having an ideal transformation as the one shown in expression (5.18), a transformation with some non-zero elements in the C_{M1} and C_{M2} matrices shown in (5.17) may be obtained. Non-zero elements in C_{M1} and C_{M2} influence the state of the control particle and may bring it to a superposition of the spin-up and spin-down state. In this case, the unitary transformation applied through this control particle would not be U_+ or U_- but a superposition of the two, leading to an error in the computation. This suggests that it is useful not only to follow the evolution of the states of the qubits, but also the evolution of the spin states of the control electrons. The quantum logic simulator works, therefore, with a state vector containing information on the states of the qubits as well as the state of the control electrons. The actual physical position of the particles (qubits and controls) on the computation chip is not taken into account in the simulator. It is rather assumed, that the physical position of the particles reflects itself in the relevant two-qubit gate parameters (e.g., in the case of SFG gates, equations (4.3) in Chapter 4) used in the analysed circuits as these depend, for example, on the distance between qubits and control particles[Rod04]. However, only to understand conceptually how the simulator works, it will be here assumed to operate with a chip as the one shown in Figure 5-2. It is assumed that the quantum register will be built out of a set of qubits which, except for qubit Q_{n-1} and Q_0 whose interaction will be described later in the paragraph, always interact with two neighbours and that there will be one control atom for each neighbouring pair of qubits. As shown in Figure 5-2, the quantum states of the register can then be defined in the following way: a given quantum state is described by a string of 0s and 1s. The bits in the odd positions, starting from the least significant bit, describe the state of the qubits, whilst the ones in the even positions describe the states of the control electrons. Figure 5-2 describes the case in which there is no interaction between qubit n-1 and qubit 0. This could correspond to a situation in which these two qubits are located too far away from each other on the chip. In this case the total number of control atoms is equal to the number of qubits -1. This configuration will be referred to as the "linear" configuration.

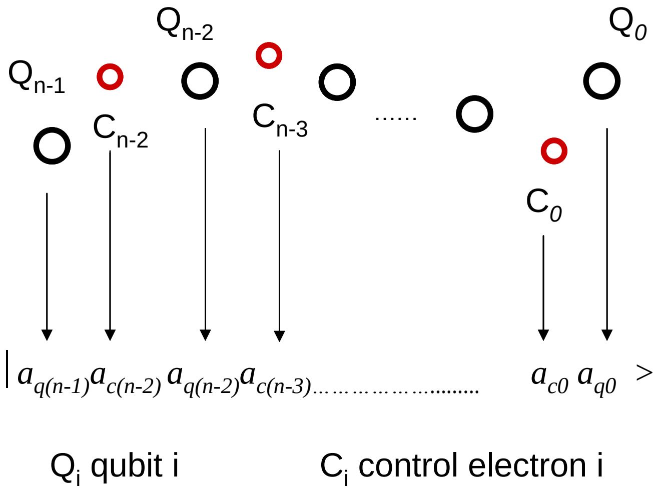

Q_{n-1}
Q_{n-2}
Q_0
C_{n-2}
C_{n-3}
C_0
……
$|a_{q(n-1)}a_{c(n-2)}\,a_{q(n-2)}a_{c(n-3)}\ldots\ldots\ldots\ldots\;a_{c0}\,a_{q0}\;>$
Q_i qubit i
C_i control electron i

Figure 5-2: Definition of the sate of the quantum register when evolution of the states of the control electrons is included in the system

In other cases, for example when considering the case of 3 qubits placed on the vertices of a triangle as shown in Figure 5-3, there can be an interaction between qubit 2 and qubit 0. In this case, which will be referred to as the "circular" configuration, the total number of control atoms is equal to the number of qubits. The configuration (linear or circular) of the qubits is specified at the beginning of the text file describing the quantum circuit.

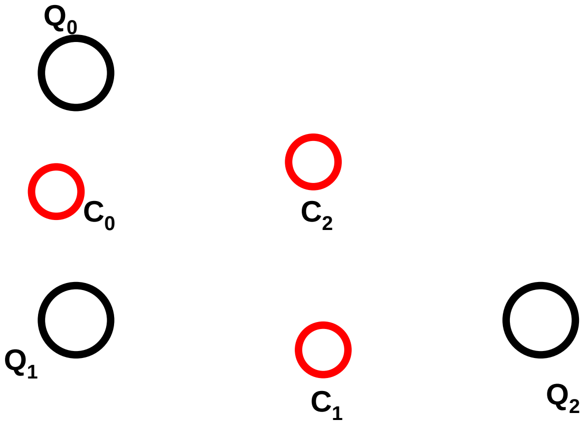


Figure 5-3: Example of circular configuration of qubits. Qubits are placed in such a way, that interaction between qubit 0 and qubit *n*-1 is possible

If, instead of only considering qubit-qubit systems, control atoms are also considered, each 2-qubit unitary transformation is then described by a 2^3x2^3 matrix, since the state of the control electron has to be taken into account, as seen, for example, from equations (5.17) and (5.18). This does not affect the main traits of the algorithms described in Section 5.1.1 for changing the state of the quantum register. Without getting into the details of the required modifications, it should only be noticed that, in terms of single-qubit operations, the same method can substantially be used. For two-qubit operations, however, instead of only extracting from the state vector the c_i coefficients corresponding to the two qubits to which the transformation has to be applied, it is now necessary to extract the coefficients which contain the information on the state of the qubits and of the corresponding control particle. These are then copied in the vector of c_i coefficients shown in expression (5.16) after having been rearranged such that their order is consistent with the 2^3x2^3 matrix which describes SFG two-qubit gates. Subsequently, the desired transformation is applied and the new c_i coefficients are copied back into the state vector, as required by the original protocol. The choice of including the control particles in the analysis of the evolution of the quantum register will increase the amount of data which will have to be dealt with since, if n qubits and n_c controls are present on the chip, a state vector of length 2^{n+n_c} will have to be processed instead of 2^n when only qubits are taken into account. On the other hand, it will give a tool which will be able to assess the penalties introduced by unwanted changes in the spin states of the control electrons which may be caused, for example, by fluctuations of the gate parameters shown in equations (4.3) of Chapter 4. As will be discussed below, it is unlikely that during an experiment these gate parameters will be obtained with perfect precision and it is therefore important to be able to model the impact of non-optimal control of the circuit.

5.1.3 Measurements

As described in Chapter 2, the act of a measurement causes the superposition of states in the quantum register to collapse randomly into one of the states building the superposition. Given a quantum register in the state:

$$\begin{aligned} |\psi\rangle &= c_0|00..00\rangle + c_1|00..01\rangle + \ldots \\ &+ c_i|a_{n-1}a_{n-2}..a_1a_0\rangle + \ldots + c_{2^n-1}|11..11\rangle \end{aligned} \qquad (5.19)$$

the probability of measuring the state $|a_{n-1}a_{n-2}..a_1a_0\rangle$ is given by $|c_i|^2$[Nie03]. This mechanism has been modelled using a uniformly distributed random number generator, with values between 0 and 1. When a measurement is to be simulated, first, a random number *r* is picked. If:

$$r \le |c_0|^2 \qquad (5.20)$$

then:

$$|\psi_{out}\rangle = |00..000\rangle \qquad (5.21)$$

is defined as the output of the measurement. Else, if:

$$r \le |c_0|^2 + |c_1|^2 \qquad (5.22)$$

then:

$$|\psi_{out}\rangle = |00..001\rangle \qquad (5.23)$$

is defined as the output of the measurement. Else, this procedure continues until:

$$r \le \sum_1^k |c_i|^2 \qquad (5.24)$$

and the corresponding state $|a_{n-1}a_{n-2}..a_1a_0\rangle$ is defined as the result of the measurement.

5.1.4 Sources of errors

5.1.4.1 Operational Errors

The matrix representations given in Table 5-1 and Table 5-2 describe the behaviour of quantum logical gates in an ideal case. But, in a practical implementation, it is unlikely that it will be possible to perform exactly the desired transformations, hence a perturbation will be introduced into the system. This problem is best viewed in the case of one-qubit gates. If the state of a qubit is described by means of the Bloch-sphere representation [Nie03] then one-qubit gates represent rotations of the state of the qubit on the surface of the Bloch-sphere. In the case of the $R_x(\theta)$ gate, for example, the initial state of the qubit is rotated by an angle θ around the x-axis of the Bloch sphere. However, in an experiment, one can expect that, due to technological reasons such as, for example, noise affecting the signals controlling the quantum computational system, a single-qubit rotation with a slightly different angle than the one expected will be implemented. Mathematically, this can be modelled by adding a random quantity ε, a perturbation, to the input angle of the gate[Sch00]:

$$R_{x-error}(\theta)=\begin{bmatrix}\cos\left(\frac{\theta+\varepsilon}{2}\right) & -i\sin\left(\frac{\theta+\varepsilon}{2}\right)\\ -i\sin\left(\frac{\theta+\varepsilon}{2}\right) & \cos\left(\frac{\theta+\varepsilon}{2}\right)\end{bmatrix} \tag{5.25}$$

All single-qubit gates can be interpreted as rotations and can be decomposed with the help of the $R_x(\theta)$, $R_y(\theta)$ and $R_z(\theta)$ matrices and a phase shift $e^{i\delta}$ [Nie03]. The Hadamard gate H, for example, can be expressed as:

$$H=\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}=e^{i\frac{\pi}{2}}R_x(\pi)R_y(\frac{\pi}{2}) \tag{5.26}$$

An operational error on a Hadamard gate can then be modelled by:

$$H_{err}=e^{i\left(\frac{\pi}{2}+\varepsilon_1\right)}R_x(\pi+\varepsilon_2)R_y\left(\frac{\pi}{2}+\varepsilon_3\right) \tag{5.27}$$

Such decompositions can be used with any single-qubit operation to introduce perturbations. Similarly, any gate which requires a variable input parameter can be perturbed by adding random fluctuations. The transformation corresponding to SFG gates depends, as reviewed previously, on physical parameters such as the magnetic field B, the strength of the interaction between qubits and controls described by J and the pulse-interleave time T. These parameters could have an offset from the desired value caused, for example, by erroneous characterisation of components used in the computation system or by unpredictable fluctuations in the devices used for the generation of the optical signal which could lead to timing-jitter on the excitation and de-excitation pulses. In our simulator, these fluctuations have been modelled through Gaussian distributions[Obe99] with standard deviation specified as an input parameter in the text file describing the quantum circuit under analysis.

5.1.4.2 Decoherence

As explained in Chapter 2, decoherence is the unwanted interaction of the environment with the quantum register. It can introduce unpredictable changes in the state of the qubits which corresponds to a loss of the stored quantum information. Computationally, this is equivalent to errors being introduced in the system.

The quantum logic simulator developed in the course of this work, currently does not include algorithms for the modelling of the impact of decoherence. The primary aim of the research was to develop compact quantum circuits suitable for the experimental demonstration of a quantum computation prototype and to study, as discussed in Chapter 6, how to best implement them within the SFG model. However, the impact of decoherence on qubits can also be modelled in a quantum logic simulator through the algorithms used, for example, by Miquel et al.[Miq96] and Devitt et al.[Dev06].

Miquel's algorithm focuses on modelling the impact of decoherence phenomena involving a transfer of energy from the qubits to the environment. The environment is modelled through so called "environmental qubits" which perturb the computational system interacting with the qubits of the quantum register. At chosen time intervals t_i, a randomly selected qubit q_i from the register interacts with an environmental qubit. The latter one always starts in the ground state $\left|0\right\rangle_{e_i}$. Defining p_1 as the probability for a qubit in the excited state $\left|1\right\rangle_{q_i}$ to remain in the excited state and p_2 =1- p_1 as the probability for the qubit q_i to decay to the ground state $\left|0\right\rangle_{q_i}$, the algorithm performs following transformation on the state of qubit q_i and the environmental qubit:

$$\begin{aligned} &\left|1\right\rangle_{q_i}\left|0\right\rangle_{e_i} \rightarrow \sqrt{p_1}\left|1\right\rangle_{q_i}\left|0\right\rangle_{e_i} + \sqrt{p_2}\left|0\right\rangle_{q_i}\left|1\right\rangle_{e_i} \\ &\left|0\right\rangle_{q_i}\left|0\right\rangle_{e_i} \rightarrow \left|0\right\rangle_{q_i}\left|0\right\rangle_{e_i} \end{aligned} \qquad (5.28)$$

Expression (5.28) can be interpreted as follows: If the qubit q_i is in its excited state, then it may remain in the excited state without interacting with the environmental qubit or it may decay to the ground state transferring its energy to the environmental qubit. Instead, if the qubit q_i is in its ground state, then there is no energy which can be transferred to the environment and no transformation is applied. The environmental qubit is discarded once these operations have been performed and the state vector is renormalized. From expression (5.28) it can be seen how the decoherence process perturbs the qubit state $\left|1\right\rangle_{q_i}$ leading to the generation of new and unwanted states.

Devitt's model simulates processes associated to T_2 with their loss of information on the relative phase difference between qubits by introducing random phase flips between them. A phase-flip probability p_{flip} is defined and, after each operational step and for each qubit, a random number rn_{flip} is picked. If $rn_{flip}<p_{flip}$, a phase-flip is applied to the qubit-state $\alpha\left|0\right\rangle+\beta\left|1\right\rangle$ by operating with a Pauli-Z matrix on it:

$$\left|qubit\right\rangle_{flip}=Z\left|qubit\right\rangle=\begin{bmatrix}1 & 0\\ 0 & -1\end{bmatrix}\begin{bmatrix}\alpha\\ \beta\end{bmatrix}=\begin{bmatrix}\alpha\\ -\beta\end{bmatrix} \qquad (5.29)$$

bringing it to the state $\alpha\left|0\right\rangle-\beta\left|1\right\rangle$.

Combining Miquel's and Devitt's models, allows one to consider, both, processes associated to T_1 and to T_2 which may perturb the computation register.

When modelling decoherence phenomena in SFG quantum computation it will be important to distinguish between decoherence affecting the qubits as opposed to decoherence perturbing the control particles.

In terms of qubits, studies have shown that, in various quantum computation proposals based on electron spin qubits in solid-state systems, i.e. with features similar to the SFG model, the decoherence effects which occur on the shortest time-scale (and which are therefore going to be the main responsible for a decrease in system performance) are T_2 processes[Tyr03]. The main cause of this perturbation arises as a result of fluctuations of the local magnetic field experienced by a qubit. This leads to fluctuations in the Larmor frequencies with which qubits evolve with respect to each other and, consequently, to uncertainties on their relative phases. One of the major sources of fluctuations in the local magnetic field comes from the non-zero nuclear spin of atoms in the substrate. In systems based on qubits carried by the electron spin of phosphorus atoms in silicon or by NV centres in diamond, this effect has been strongly reduced by exploiting purified substrates with a very high percentage of atoms with 0 nuclear spin achieving, for both systems, decoherence times in the order of ~ms[Tyr03,Bal09]. Hence, because the SFG model envisages qubit structures similar to the ones analysed in [Tyr03] or may be compatible with the diamond NV proposal, the dominant decoherence effect perturbing the qubits is likely to be dephasing of the qubits which can be simulated with the phase-flip model described, for example, by Devitt et al..

In terms of control particles, the main effect which will perturb the SFG scheme will be the unwanted relaxation to the ground state of the excited state of the control particle while it mediates an interaction between two qubits. In theory, as discussed in the previous chapter, the SFG proposal requires that, for a desired quantum gate to be applied between two qubits, its corresponding control particle remains in an excited state for a specific time T_{ideal}. If, however, the control particle relaxes to the ground state after a time T_{err} shorter that T_{ideal}, then the two qubits interact for a time shorter than the desired one and are therefore subject to an incomplete transformation. In a recent study on the lifetime of the excited state of phosphorus atoms in silicon it was shown that one of the main contributions to the decoherence of the excited state is decay by emission of a phonon[Vin08]. This mechanism is a T_1 process which involves an exchange of energy between the computation register and the substrate and can be simulated with a model similar to the one proposed by Miquel et al.. More specifically, a phenomenological model for simulating relaxation of the control particle could be the following: Firstly, a probability distribution p_{T1} for the perturbed pulse interleave time T_{err} has to be defined. This probability distribution will be used, in conjunction with a random number generator, to obtain a perturbed pulse interleave time T_{err} whenever a relaxation event has to be modelled. The values returned by p_{T1} should reflect as close as possible the time-scales characterising this relaxation phenomenon and should therefore be also function of the specific T_1 value for a given experiment. Hence, for $T_{ideal}<T_1$, p_{T1} should return with a higher probability T_{err} values larger than T_{ideal}, while for $T_{ideal}>T_1$, values of $T_{err}<T_{ideal}$ should be returned with higher probability. The exact shape of p_{T1} will depend on the specific materials used as control particles and

substrates (since these will determine the physical processes responsible for the relaxation event). In general terms, it should be noticed that an exponential behaviour in time characterised by the decoherence time T_1 is often associated to decoherence phenomena[Nie03] and may therefore be a convenient starting point for defining p_{T1}. Once this distribution has been defined and a possible decoherence event has to be modelled for a given two-qubit gate described through a pulse interleave time T_{ideal}, a random number is picked in order to determine, through p_{T1}, the perturbed interaction time T_{err}. If the resulting T_{err} is longer than T_{ideal} then it is assumed that the decoherence event would take place after the transmission of the de-exciting pulse which returns the control particle to the ground state, meaning that the ideal transformation can be implemented before decoherence perturbs gate operation. Hence, the ideal two-qubit gate is applied to the qubits in this case. If, however, T_{err} is shorter than T_{ideal} it is assumed that the control particle relaxes back to the ground state before the arrival of the de-exciting pulse and the unitary transformations describing the interaction between qubits and controls are evaluated using T_{err} in the relevant equations instead of T_{ideal}. Moreover, this model should also take into account that, in case an unwanted relaxation event of the control electron takes place prior to the transmission of the de-exciting pulse, then, since the de-exciting pulse will nevertheless be transmitted on the chip, this will actually lead to the renewed excitation of the control electron and, consequently, to the unwanted interaction of the qubits until the control electron relaxes back to its ground state. This further perturbation can be modelled as a second (un-scheduled) SFG interaction with interaction time T_{err2} evaluated again using p_{T1} and a random number generator.

Although the implementation of these protocols fell outside the scope of the work described in the thesis, a detailed analysis on how decoherence may perturb the performance of the circuits presented in Chapter 6 is important for obtaining a more complete picture on the potential of the SFG model. Hence, the integration of the quantum logic simulator with algorithms for simulating the impact of decoherence is important in future work.

5.1.5 Output data

The main output data of the simulator is the state vector $|\psi_{out}\rangle$ prior to the final measurement which would give the desired result of the computation. This vector can be used:

1) To evaluate the probability distribution of the output states taking the squared modulus of the probability amplitudes

2) To compare different output vectors through the computation of the fidelity [Obe99,Sch00] which, given two output vectors $|\psi_{out-1}\rangle$ and $|\psi_{out-2}\rangle$, is defined as:

$$Fidelity = \left|\langle\psi_{out-1}|\psi_{out-2}\rangle\right|^2 \qquad (5.30)$$

The fidelity is equal to 1 if the two output state vectors are equivalent (parallel) and 0 if the two vectors are orthogonal. It can be used, for example, to evaluate the impact of impairments on a given circuit implementation for increasing amount of perturbation. Alternatively, given a set of different quantum circuits which solve the same quantum algorithm, the fidelity can be used to compare the tolerance of the various circuits towards a specific impairment.

5.1.6 Using the simulator: an example

An example of the simulator operation is given here to show how a text file is written. The simulated circuit is a C-NOT gate between two qubits of a 3-qubit quantum register *q2, q1, q0*. The C-NOT gate is not implemented directly, but through the equivalent circuit shown in Figure 5-4 which exploits two Hadamard gates *H* and a controlled-phase gate *CP*.

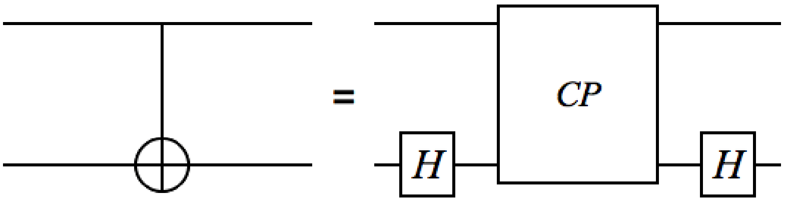


Figure 5-4: Equivalence circuit for the C-NOT and the controlled-phase gate

As shown in Chapter 4, a controlled phase gate can be approximated using an SFG gate with *M*=1584 and *N*=2177. The quantum register is initialised so that all particles, both, qubits and control atoms, are in the spin-up state, mathematically equivalent to them being in the $|0\rangle$-state and qubits are considered to be in the circular configuration, as in

Figure 5-3. First, a single qubit rotation is applied to qubit *q2* in order to bring it to the $|1\rangle$-state. Then, the equivalent C-NOT circuit shown in Figure 5-4 is applied with qubit *q2* as the control qubit and *q0* as the target qubit. Since, after the rotation, *q2* is in the $|1\rangle$-state and *q0* in the $|0\rangle$-state, the C-NOT gate will bring qubit *q0* to $|1\rangle$. Figure 5-5 shows the corresponding complete circuit.

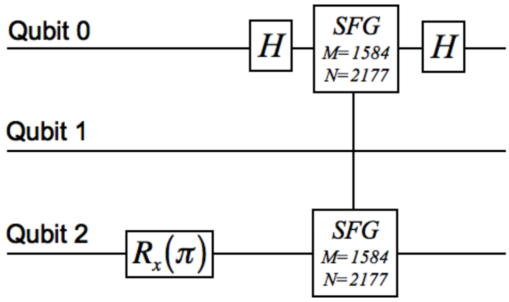
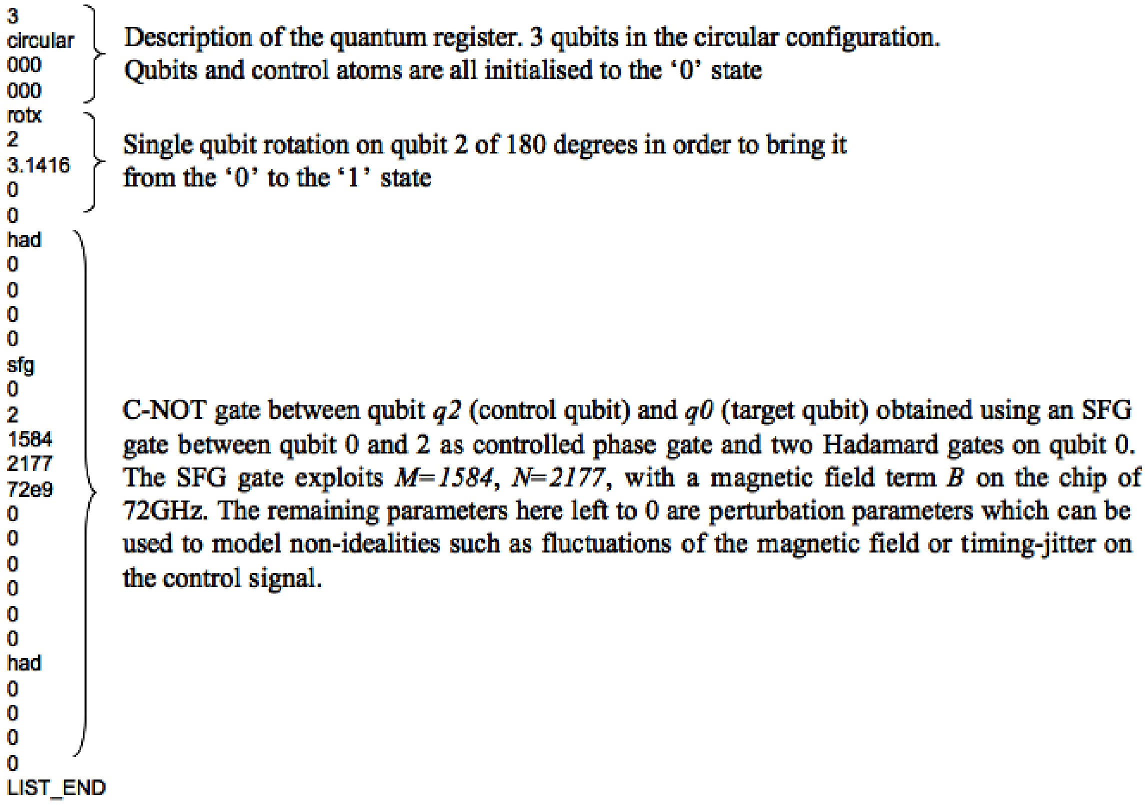


Figure 5-5: Circuit simulated for the demonstration of a C-NOT gate implemented using an SFG gate

The text file for the simulator describing the circuit shown in Figure 5-5 is:



```
3
circular
000
000
rotx
2
3.1416
0
0
had
0
0
0
0
sfg
0
2
1584
2177
72e9
0
0
0
0
0
0
had
0
0
0
0
LIST_END
```

Description of the quantum register. 3 qubits in the circular configuration. Qubits and control atoms are all initialised to the ‘0’ state

Single qubit rotation on qubit 2 of 180 degrees in order to bring it from the ‘0’ to the ‘1’ state

C-NOT gate between qubit *q2* (control qubit) and *q0* (target qubit) obtained using an SFG gate between qubit 0 and 2 as controlled phase gate and two Hadamard gates on qubit 0. The SFG gate exploits *M=1584*, *N=2177*, with a magnetic field term *B* on the chip of 72GHz. The remaining parameters here left to 0 are perturbation parameters which can be used to model non-idealities such as fluctuations of the magnetic field or timing-jitter on the control signal.

The input (red) and output (blue) probability distribution of the quantum register, are shown in Figure 5-6, taking into consideration only the state of the qubits and not of the control atoms.

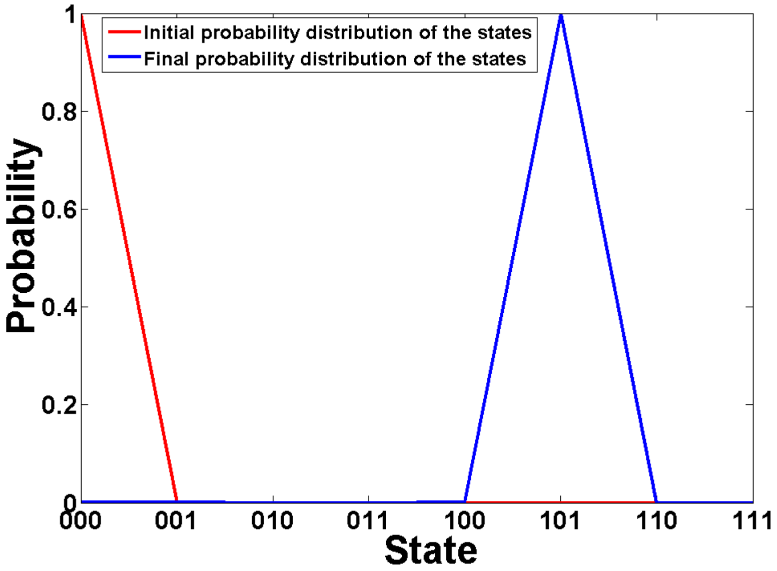


Figure 5-6: Input and output probability distribution for the C-NOT equivalent circuit

Since all qubits of the quantum register are initialized to the $|0\rangle$-state at the beginning, only the state $|000\rangle$ has a finite probability of being measured in the input distribution. Conversely, with *q2* being brought to the $|1\rangle$-state by the rotation and *q0* brought to $|1\rangle$ by the C-NOT gate, the only state which has a non-zero probability of being measured in the output state is $|101\rangle$.

Summarising, the main features of a quantum logic simulator developed during the project have been described. This tool was specifically designed for analysing circuits based on SFG gates since it not only simulates the evolution of the qubits, but also models the state of the control particles. This feature is important for the analysis of circuits based on the SFG model, since an unwanted change in the state of a control particle can subsequently lead to perturbations in the corresponding SFG gates. The quantum logic simulator has been used to analyse the performance of circuits suitable for the experimental demonstration of an SFG quantum computation prototype and the results obtained through this analysis are presented in Chapter 6.

However, to analyse a quantum logic circuit, one first needs to design it. Partly, this task has been solved in this work by means of an automated quantum circuit design algorithm based on a genetic programming approach which is described below.

5.2 A genetic programming algorithm for quantum circuit design

As described in the Chapter 2, designing the quantum circuit which implements a desired quantum computation, requires to find a corresponding well-defined sequence of single- and two-qubit gates (out of the set of quantum gates allowed by the chosen technology), which implements the unitary operator U_{comp} describing the computation. The final quantum circuit depends, therefore, on the set of available gates. Physical constraints, typically the most important being the impact of decoherence, may influence the design process. Often this translates into the search for the circuit with the least number of gates.

Mathematically, the quantum circuit design problem can be restated in terms of a matrix decomposition. Given the unitary transformation U_{comp} describing a quantum computation and given a set of matrices $\{U_1, U_2, U_3 \ldots\}$ describing the available gates, designing a quantum circuit is equivalent to finding a decomposition of U_{comp} comprising only gates U_i out of the available set. For example:

$$U_4 \cdot U_1 \cdot U_3 \cdot U_6 \cdot U_2 \cdot U_6 = U_{comp} \qquad (5.31)$$

In the work described in this thesis, the aim was to derive circuits compatible with SFG technology for a three-qubit refined Deutsch-Jozsa algorithm[Col98]. Analytical methods have been previously proposed for solving this problem with other quantum computational systems. In [Kim00b], for example, an algorithm based on generator expansion was proposed which was then used to implement a three-qubit refined Deutsch-Jozsa algorithm on a quantum computer based on NMR[Kim00a]. However, the generator expansion technique addresses the quantum circuit design problem by analysing the Hamiltonian which describes the interactions between the qubits and the computation one wishes to perform. In the SFG proposal the situation is complicated by the presence of the control atoms which make it difficult to apply the generator expansion technique to this model. Also, the generator expansion algorithm exploits C-NOT gates to reduce interactions between more than two qubits to operations based on two-qubit gates only, as shown in Figure 5-7[Kim00b].

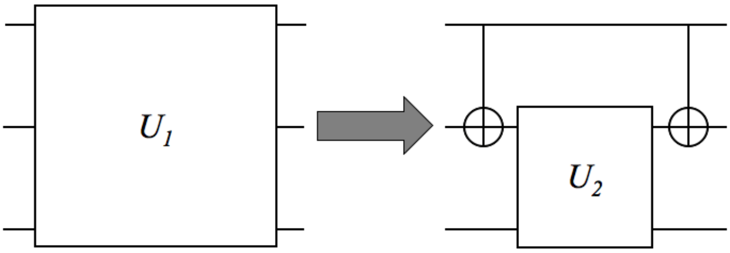


Figure 5-7: Reduction of three-qubit interactions to two-qubit interactions[Kim00b]

This places a constraint on the design procedure: the C-NOT gate, or an equivalent gate such as the controlled-phase gate, must be part of the quantum gate library used for the decomposition process. As mentioned in Chapter 4, it is possible to produce SFG gates which approximate controlled-phase gates and these approximations differ in their level of precision and in their gate operation times. However, as will be discussed in the next chapter, there are other entangling SFG gates which have gate computation times shorter than those of the approximated controlled-phase gates. Hence, in order to investigate strategies for designing efficient quantum logic circuits with SFG gates and to compare how circuit topologies may change depending on the entangling gates used during the decomposition process, a more flexible design procedure compared to the generator expansion algorithm, one which does not impose a priori a specific set of gates, may be more desirable and useful. A numerical method which incorporates this flexibility is the genetic programming algorithm adapted to quantum circuit design proposed by Williams and Gray[Wil99]. Their model is based on following idea: suppose one has a unitary transformation U_{comp} which describes a quantum computation one wishes to perform on a quantum register. Further, suppose one has a specific set of one- and two-qubit gates, described mathematically by U_1, U_2, U_3, $U_4 \ldots U_i$, which one is able to apply to the quantum register. Williams and Gray start their design process by arbitrarily creating an initial population of circuits. Each circuit U_{circ} comprises a random sequence of one- and two-qubit gates out of the available set:

$$U_{circ} = U_k \cdot U_m \cdot U_k \cdot U_l \cdot U_n \cdot \ldots\ldots : U_n \quad \{k,l,m,n\} \in \{1,2..i\} \qquad (5.32)$$

A fitness parameter f_{fit} which quantifies how well a circuit implements the desired transformation U_{comp}, is then evaluated for each element of the population. After that, the circuits are sorted according to their fitness value. Once the initial population has been built, following protocol is implemented:

1) *Parents* which will breed the circuits of the next generation are selected from the population. The selection procedure is random, although circuits with a higher value of fitness have a higher probability of being picked.

2) The next generation is created by crossover and mutation of the selected parents. In crossover, a new circuit is built by connecting two random fractions of circuits of two parents as shown in Figure 5-8.

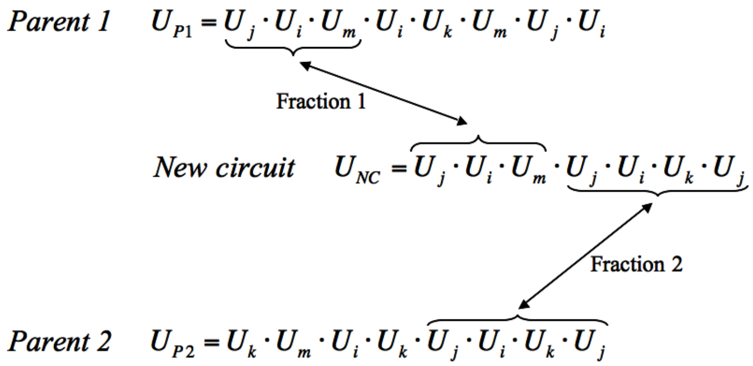

Parent 1
$U_{P1} = \underbrace{U_j \cdot U_i \cdot U_m} \cdot U_i \cdot U_k \cdot U_m \cdot U_j \cdot U_i$
Fraction 1
New circuit
$U_{NC} = \overbrace{U_j \cdot U_i \cdot U_m} \cdot \underbrace{U_j \cdot U_i \cdot U_k \cdot U_j}$
Fraction 2
Parent 2
$U_{P2} = U_k \cdot U_m \cdot U_i \cdot U_k \cdot \overbrace{U_j \cdot U_i \cdot U_k \cdot U_j}$

Figure 5-8: Generation of a new circuit through crossover

Conversely, mutation perturbs the circuit represented by a parent by typically inserting a random gate, deleting a random gate or perturbing an existing gate as will be described later in more detail.

3) Once a new generation is formed, the fitness of each circuit is assessed and the population is sorted according to the fitness value of its circuits.

4) If at least one circuit in the new population has a fitness value which reaches a desired threshold or, in order to avoid excessive computation time, if a maximal number of iterations has been reached, the algorithm terminates, otherwise the procedure returns to point 1).

The idea behind genetic programming is that, by randomly composing or mutating circuits (giving a stronger weight to ones with a high fitness), generations of better-performing circuits might subsequently be built.

Using genetic programming, Williams and Gray developed circuits for the teleportation problem[Wil99]. As reviewed in [Spe04], this method was applied to other problems including the one-qubit Deutsch-Jozsa algorithm. Other studies have analysed the impact on the search efficiency of different strategies for integrating fitness and cost functions when assessing the quality of a circuit [Luk03] or the influence of alternative selection strategies [Lei04]. While Williams and Gray's original work was based on finding deterministic circuits, other researchers have obtained important results limiting the search to probabilistic circuits, i.e. circuits which would yield the correct result with a probability of success p>0.5, as shown, for example, in [Spe04,Mas04]. In most of these studies, the quantum logic gates used for the design process were ideal and technology-independent. However, in the work here presented, the genetic programming algorithm was specifically developed to operate with SFG gates, in order, as will be further discussed in the next chapter, to design quantum logic circuits suitable for the experimental demonstration of a prototype based on this technology.

5.2.1 Implementation of the genetic programming algorithm

The design tool presented here was built specifically for designing circuits for a potential three-qubit quantum computation experiment. The qubits were assumed to have a physical distribution such that each of the three can interact with the others as shown in Figure 5-9.

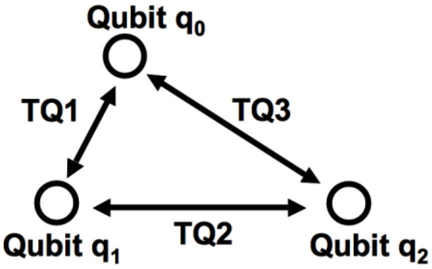


Figure 5-9: Three-qubit scheme assumed for the simulations

The set of gates used in the quantum circuit design algorithm comprises three two-qubit entangling gates TQ1, TQ2 and TQ3 representing, respectively, a two-qubit gate between qubit q_0 and q_1, q_1 and q_2, q_0 and q_2 (Figure 5-9). These entangling gates could, for example, be any of the gates commonly used in literature such as the controlled-phase gate or the C-NOT gate, or SFG gates. In terms of single-qubit gates, the $R_z(\theta)$ and $R_x(\theta)$ rotation operators and a phase shift operator $Phi(\theta)$[Nie03]:

$$R_x(\theta)=\begin{bmatrix}\cos\left(\frac{\theta}{2}\right) & -i\sin\left(\frac{\theta}{2}\right)\\ -i\sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right)\end{bmatrix};\ R_z(\theta)=\begin{bmatrix}e^{-i\frac{\theta}{2}} & 0\\ 0 & e^{i\frac{\theta}{2}}\end{bmatrix};\ Phi(\theta)=e^{-i\frac{\theta}{2}}\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix} \quad (5.33)$$

were taken into account as these, together with two-qubit entangling gates, form a universal set of quantum logic gates [Nie03,Deu95]. A circuit of L quantum gates is specified through a 2xL matrix in which the first row memorises the type of gate and on which qubit(s) it operates while the second row stores the angles of single-qubit rotations. The following codification has been used with respect to the gate labels used in the first row of the matrix:

- 1,2 and 3 respectively describe the entangling gates TQ1, TQ2 and TQ3 shown in Figure 5-9. The matrices describing the entangling gates TQ1, TQ2 and TQ3 are given as input to the algorithm.

- 4,5, or 6 describe an R_z rotation on qubits 0,1 or 2.

-7,8 or 9 an R_x rotation on qubits 0,1, or 2.

-10,11 or 12 a phase shift on qubits 0,1 or 2.

The gates are stored in the *2xL* matrix sequentially. Figure 5-10 shows an example of a 2xL circuit matrix used in the algorithm and the corresponding circuit.

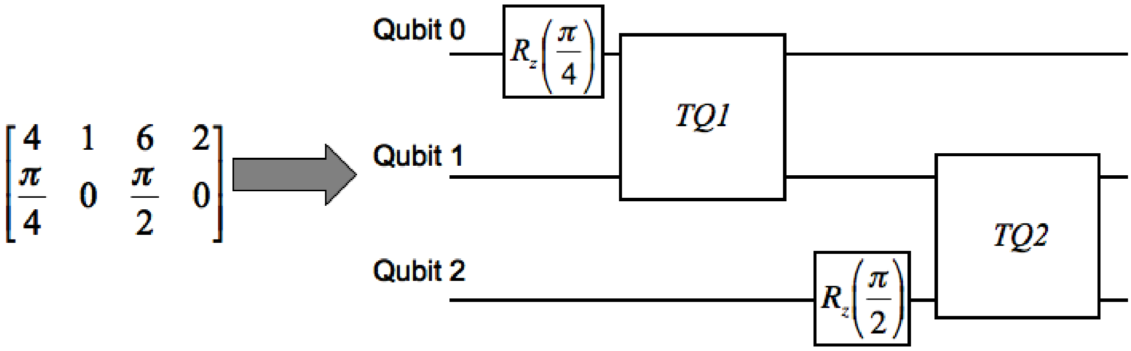

$\begin{bmatrix} 4 & 1 & 6 & 2 \\ \frac{\pi}{4} & 0 & \frac{\pi}{2} & 0 \end{bmatrix}$
Qubit 0
Qubit 1
Qubit 2
$R_z\left(\frac{\pi}{4}\right)$
TQ1
$R_z\left(\frac{\pi}{2}\right)$
TQ2

Figure 5-10: Example of 2x*L* matrix representation used in the genetic programming algorithm and the corresponding circuit.

5.2.2 Fitness evaluation

The average fidelity *AF* of the transformation U_{circ} produced by a given circuit, with respect to the ideal desired transformation U_{comp} is used as the fitness parameter[Nie02,For02,Rod04]:

$$AF=|Tr(U^{\dagger}_{comp}U_{circ})/2^{Nq}|^2 \qquad (5.34)$$

where *Nq* is the number of qubits in the system. *AF* is equal to 1 when U_{circ} implements U_{comp} exactly (up to an irrelevant phase difference), while lower values are returned in case of imperfect implementation. The average fidelity has been already used to evaluate the fitness of a circuit, see, for example, [Din08,Rei05], although in these cases, *AF* was only one factor in a multi-parameter fitness function. Here, it was preferred to implement a fitness function only based on the average fidelity in order to avoid having too many parameters biasing the search process and, possibly, not understanding how the various parameters influence the convergence of the algorithm.

5.2.3 Generation of the initial population

The dimension of the population in a design process is defined at the beginning and labelled PopL. To create the initial population of circuits, PopL random sequences of gates are generated. The length of the circuits is not fixed. However, in order to avoid solutions corresponding to excessively long quantum circuits becoming dominant in a generation, a maximum allowed number of two-qubit gates per circuit (TQmax) has been introduced in this work. All functions in the algorithm are designed such that evolutions of circuits never exceed TQmax. Also, “cleaning” functions are exploited which condense repetitions of adjacent single-qubit gates of the same type into a single gate by adding up their rotation angles. Once the pool of circuits has been generated, the fitness of all circuits is assessed and the population is sorted according to the fitness value.

5.2.4 Mutation and crossover functions

Mutation of a circuit requires the quantum gate sequence to be altered. Four different mutation functions have been implemented in this work: removal of a random gate from the gate sequence, insertion of a random gate in the gate sequence, exchange of a random gate in the circuit with a random gate from the available set and perturbation of a random gate in the circuit. In the latter case, in case of two-qubit gates, the qubits on which the selected gate is operating are randomly changed while in case of single-qubit operations a new angle θ for the rotation is picked. When mutation is applied to a circuit, one out of the four possible functions is randomly selected with equal selection probability and implemented on the gate sequence.

As shown in Figure 5-8, given two parents, crossover is implemented by connecting two random fractions of the parents to form a new circuit. Supposing that Parent1 has length L1 and Parent2 length L2, the fraction forming the initial part of the circuit is obtained by selecting a random number k_1 in the range between 1 and L1, and taking the first k_1 gates of Parent1. The fraction forming the final part of the circuit is obtained picking a random number k_2 between 1 and L2 and taking the gates from gate k_2 to gate L2 of Parent2.

5.2.5 Selection mechanism

Given a population, parents for, both, mutation and crossover have to be selected at each round to breed the circuits of the next generation. The selection mechanism has to be random, with a higher probability of being selected given to circuits with a higher fitness value. In the genetic programming tool presented here stochastic universal sampling has been used which is a sampling method commonly used in genetic programming (see, for example, [Luk03]). An example of this sampling method is shown schematically in Figure 5-11 for a population of 8 elements out of which N elements must be selected. The population is sorted according to the fitness value and the structure shown in Figure 5-11 is formed in which, for each element, a slot proportional to its fitness value is associated. Then, a vector of N pointers equally spaced by $\frac{F_{TOT}}{N}$ (with F_{TOT} being the sum of all fitness values) is generated. Finally, a random number $\Delta\frac{F_{TOT}}{N}$ between 0 and $\frac{F_{TOT}}{N}$ is picked. This number randomly shifts the vector of pointers and how it is superimposed over the slots representing the population elements. In the example shown in Figure 5-11, for instance, using stochastic universal sampling with N=9 leads to element 1 being selected 3 times as a parent, element 2 twice and elements 3,4,5 and 7 once.

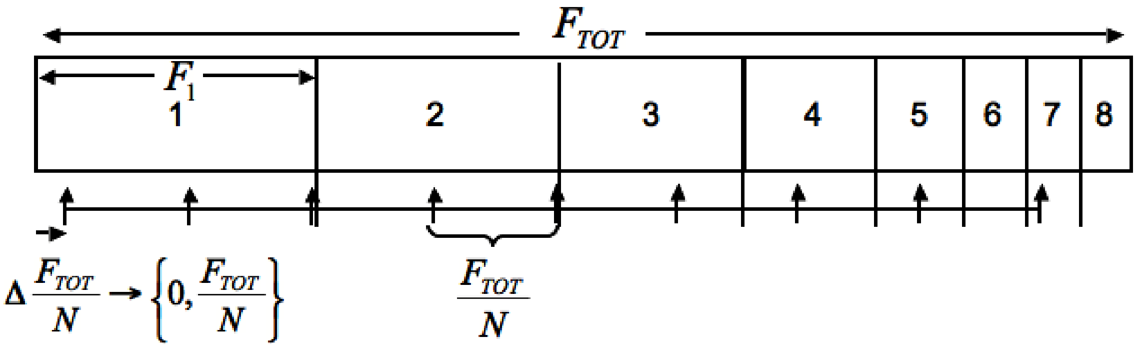

F_{TOT}
F_1
1
2
3
4
5
6
7
8
$\Delta\frac{F_{TOT}}{N} \rightarrow \left\{0, \frac{F_{TOT}}{N}\right\}$
$\frac{F_{TOT}}{N}$

Figure 5-11: Schematic representation of the SUS scheme

In each round, the number of parents needed for mutation and crossover is fixed and known. These quantities are defined through two parameters, CrossProb and MutProb, given as input at the beginning of the design process, which represent, respectively, the fraction of population which will be bred through crossover and mutation. At the beginning of each round 2· CrossProb·PopL circuits are selected as parents for crossover (two parents are needed for each new circuit) while MutProb·PopL circuits are selected for mutation. Finally, in order to avoid the loss of the best solutions obtained during an algorithm cycle, moderate elitism, meaning the direct transfer of the fittest elements in the population to the next generation without applying crossover or mutation [Step08], is exploited in the design process. Defining the parameter ElitProb, elitism is implemented selecting the best ElitProb·PopL circuits in the population and transferring them to the next generation. In the work here presented, CrossProb typically spanned between 0.2 and 0.5, MutProb between 0.5 and 0.8 while ElitProb assumed values around 0.01. In most of the design processes PopL=5000 was used. The quantum circuits obtained with this design algorithm for a three-qubit quantum computation register based on the SFG model will be presented in the next chapter.

5.3 Summary

The main features of two numerical tools developed throughout the project have been described: a quantum logic simulator and an automated quantum circuit design algorithm based on genetic programming.

The quantum logic simulator takes as input the initial state of a quantum register and applies a sequence of unitary operators corresponding to a quantum circuit one wants to analyse in order to compute the output state of the computation. It also allows one to model the non-ideal behaviour of the quantum logic gates in a circuit. This numerical tool can be used to test whether or not a circuit correctly solves a given algorithm or to compare the performance of different circuits which solve the same problem in order to understand which topology might be more resistant to a certain class of errors. The quantum gate library of the simulator can be easily updated and contains a variety of different gates: from abstract, non-implementation dependent gates such as the C-NOT, to models of physical gates such as the SFG gate or the two qubit gate describing interactions in NMR quantum computation.

The second numerical tool addresses the issue of quantum circuit design. Given a unitary matrix U_{comp}, describing the quantum computation one wants to implement, to actually perform that computation with a physical quantum computer requires to find a sequence of single- and two-qubit gates (out of the set of gates which can be produced within the chosen technology) which generates U_{comp}. The automated quantum circuit design algorithm discussed above solves this problem by means of a genetic programming approach. This algorithm has been chosen because of its flexibility in terms of the two-qubit quantum gates which can be used in the decomposition process and has been specifically tailored for the case of a three-qubit quantum computational system based on SFG gates.

In the next chapter, the results of how these tools have been used for the design and analysis of quantum logic circuits suitable for the experimental demonstration of a three-qubit quantum computer prototype based on SFG gates are described.

5.4 References

[Bal09] G.Balasubramanian, P.Neumann, D.Twitchen, M.Markham et al.: “Ultralong spin coherence time in isotopically engineered diamond”, *Nature Materials*, vol.8, pp. 383-387, (2009)

[Bar95] A. Barenco et al.: “Elementary gates for quantum computation”, *Physical Review A*, Vol. 52, No. 5, pp. 3457-3467, (1995)

[Col98] D. Collins, K.W. Kim et al.: “Deutsch-Jozsa algorithm as a test for quantum computation”, *Physical Review A*, vol.58, pp 1633-1636, (1998)

[Deu95] D.Deutsch, A.Barenco et al.: “Universality in quantum computation”, *Proceedings of the Royal Society of London, Series A, Mathematical and Physical Sciences*, vol.449, pp.669-677, (1995)

[Dev06] S.J.Devitt, A.G.Fowler, L.C.L.Hollenberg: “Robustness of Shor’s algorithm”, *Quantum Computation & Information*, vol.6, pp616, (2006)

[Din08] S.Ding, Z.Jin, Q.Yang: “Evolving quantum circuits at the gate level with a hybrid quantum-inspired evolutionary algorithm”, Soft Computing, vol.12, pp.1059-1072, (2008)

[Eke96] A. Ekert, R. Jozsa: “Quantum computation and Shor’s factoring algorithm”, *Review of Modern Physics*, vol.68, No.3, pp. 733-753, (1996)

[For02] E.M.Fortunato, M.A.Pravia et al.: “Design of strongly modulating pulses to implement precise effective Hamiltonians for quantum information processing”, *Preprint quant-ph/0202065*, (2002)

[Jon98] J. A. Jones, M. Mosca: “Implementation of a quantum algorithm on a nuclear magnetic resonance quantum computer”, *Journal of Chemical Physics*, vol.109, pp.1648-1653, (1998)

[Kim00a] J.Kim, J.S.Lee, S.Lee, C.Cheong: “Implementation of the refined Deutsch-Jozsa algorithm on a three-bit NMR quantum computer”, *Physical Review A*, vol.62, Article number 022312, (2000)

[Kim00b] J.Kim, J.S.Lee, S.Lee: “Implementing unitary operators in quantum computation”, *Physical Review A*, vol.61, Article number 032312, (2000)

[Lei04] A.Leier, W.Banzhaf: “Comparison of selection strategies for evolutionary quantum circuit design”, *Lecture Notes in Computer Science, Proceedings of the 2004 Genetic and Evolutionary Computation Conference (GECCO 2004)*, vol. 3103, pp.557-568, (2004) [Luk03] M.Lukac, M.Perkovsky et al.: “Evolutionary approach to quantum and reversible circuits synthesis”, *Artificial Intelligence Review*, vol.20, pp.361-417, (2003)

[Mas04] P.Massey, J.A.Clark, S.Stepney: “Evolving quantum circuits and programs through genetic programming”, Lecture Notes in Computer Science, Proceedings of the 2004 Genetic and Evolutionary Computation Conference (GECCO 2004), Vol. 3103, pp.569-580, (2004)

[Miq96] C. Miquel, J.P. Paz, R. Perazzo: “Factoring in a dissipative computer”, *Physical Review A*, vol.54, No. 4, pp. 2605-2613, (1996)

[Nie02] M.Nielsen: “A simple formula for the average gate fidelity of a quantum dynamical operation”, *Preprint quant-ph/0205035*, (2002)

[Nie03] M.A.Nielsen, I.L.Chuang: “Quantum Computation and Quantum Information”, Cambridge University Press, (2003)

[Obe99] K.M. Obenland, A.M.Despain, “Simulating the effect of decoherence and inaccuracies on a quantum computer”, *in Quantum Computing and Quantum Communications, Book Series: Lecture Notes in Computer Science*, vol. 1509, pp.447-459, (1999)

[Pet05] J.R.Petta, A.C.Johnson et al.: “Coherent manipulations of coupled electron spins in semiconductor quantum dots”, *Science*, vol.309, p.2180-2184, (2005)

[Rei05] T.Reid: “On the evolutionary design of quantum circuits”, Master’s Thesis, University of Waterloo, Canada, (2005)

[Rod04] R.Rodriquez, A.J.Fisher, P.T.Greenland, A.M.Stoneham: “Avoiding entaglement loss when two qubit gates are controlled by electronic excitation”, *Journal of Physics: Condensed Matter*, vol.16, pp.2757-2772,(2004)

[Sch00] F. Schürmann: “Interactive Quantum Computation”, Master Thesis, submitted in September 2000, University of New York at Buffalo.

[Sch03] F. Schmidt-Kaler, H.Häffner et al.: “Realization of the Cirac-Zoller controlled-NOT quantum gate”, *Nature*, vol.422, pp.408-411, (2003)

[Spe04] L.Spector: “Automatic Quantum Computer Programming: A Genetic Programming Approach”, Kluwer Academic Publishers, (2004) [Step08] S.Stepney, J.A.Clark: “Searching for quantum programs and quantum protocols”, *Journal of Computational and Theoretical Nanoscience,* vol.5, pp.942-969, (2008)

[Tyr03] A.M.Tyryshkin, S.A.Lyon et al.: “Electron spin relaxation times of phosphorus donors in silicon”, *Physical Review B,* vol.68, Article number 193207, (2003)

[Wil99] C.P. Williams, A.G. Gray, “Automated design of quantum circuits”, *in Quantum Computing and Quantum Communications, Book Series: Lecture Notes in Computer Science*, vol. 1509, pp.113-125, (1999)

Chapter 6 Design of quantum circuits based on SFG technology

This chapter analyses the design of quantum circuits suitable for the experimental demonstration of a three-qubit quantum computation prototype based on the SFG model described in Chapter 4. The algorithm used for this study is a refined version of the Deutsch-Jozsa algorithm[Deu92,Cle98,Col98], an algorithm commonly used for the experimental demonstration of small-scale quantum computation prototypes (see [Chu98, Jon98, Kim00a], for example).

The aim of the work was to find circuits with a short computational time in order to reduce the probability of errors accumulating along the computational path. To achieve this, various design steps have been implemented. A first circuit implementing the refined Deutsch-Jozsa algorithm has been obtained adapting a circuit developed for an NMR quantum computational system in [Kim00a] to one compatible with controlled-phase gates implemented with the SFG model, while the remaining circuits presented in this chapter have been obtained by means of the automated quantum circuit design algorithm based on the genetic programming approach which was presented in Chapter 5. This tool has been used here to, first, derive quantum logic circuits based on ideal controlled-phase gates, comparing the obtained solutions to the ones presented by Kim et al.[Km00a]. Then, the automated quantum circuit design approach was used to analyse how the performance of the circuits solving the refined Deutsch-Jozsa algorithm varied when the ideal controlled-phase gates were substituted by SFG gates, focusing on exploiting SFG gates with decreasing gate computation time. Specifically, three different sets of SFG gates were tested. The first two sets implemented approximations of controlled-phase gates. As will be described in more detail below, the first set approximated controlled-phase gates with very high accuracy at the expense of long gate computation times, while the second set exploited gates with gate computation times around 30 times faster than those of the first set, although approximating the controlled-phase gates with less precision. Remembering, as seen in Chapter 4, that a variety of different entangling gates can be obtained within the SFG model, the third set comprised SFG gates which produce arbitrary entangling gates (meaning gates different from the ones typically used in literature such as, for example, controlled-phase gates, C-NOT gates or $\sqrt{SWAP}$ gates) with gate computation times shorter than those of the other two sets.

All the circuit presented in this chapter have been derived assuming perfect control over the quantum gates parameters. However, during an experiment, this is not going to be the case and it is important to understand how the performance of the circuits may change because of non-optimal control of the quantum gates. Hence, focusing on one of the presented solutions, the change in circuit's performance due to fluctuations of some specific gate parameters is also discussed.

Finally, because SFG quantum logic gates are implemented by exciting control particles through optical pulses (which means that the implementation of a quantum logic circuit requires to produce sequences of optical pulses which are then transmitted on the quantum computation chip) the last part of this chapter is dedicated to a discussion on optical systems which may be suitable for generating the optical signals necessary to implement the presented quantum circuits.

6.1 Deutsch's problem and the refined Deutsch-Jozsa algorithm

Suppose an oracle implements an *n*-bit function $f_n(x)$ which can be either constant, i.e. always returns either 1 or 0 for any input value *x*, or balanced, i.e. returns 1 for exactly half of all possible input values and 0 for the remaining ones. Deutsch's problem asks how many queries to the oracle are necessary to determine whether the given function is constant or balanced[Cle98]. With a classical approach one needs, in the worst case, $2^{n-1}+1$ (half of all possible input values plus one) queries since, if the first 2^{n-1} values return the same output, say 1, one still needs one more query to define the nature of the function[Deu92]. If the function implemented by the oracle is constant the next returned value will be again 1 whereas it will be 0 in the case of a balanced one. However, as described in [Nie03], Deutsch showed that using a quantum computational approach, exploiting the parallelism naturally implemented by such systems, only one query to the oracle is necessary. Further, developing a refined version of the algorithm, Collins et al. showed that the circuit shown in Figure 6-1 correctly solves the problem [Col98]:

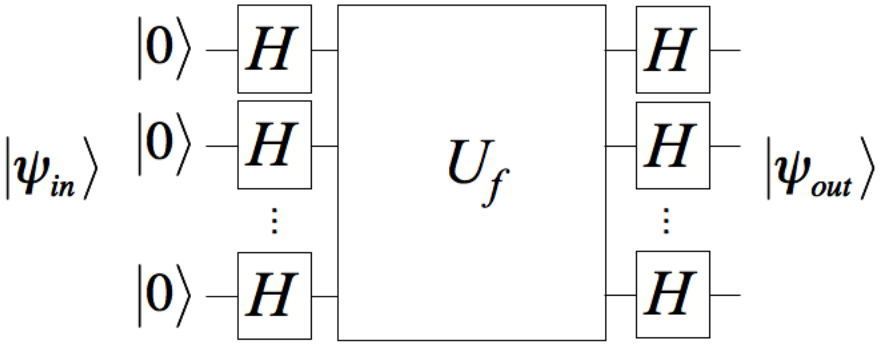


Figure 6-1: Quantum circuit solving the refined Deutsch-Jozsa algorithm

where the Hadamard gate *H* is:

$$H = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{6.1}$$

and U_f is an operator which implements the function defined by the oracle. The circuit starts with an input state $|\psi_{in}\rangle$ with all *n* qubits in the $|0\rangle$-state. Then, the Hadamard gates load the qubits with an equal superposition of all possible 2^n values which such a quantum register can represent. The operator U_f is then applied to the 2^n input values while the final set of Hadamard gates makes these values interfere. A measurement of the output-state $|\psi_{out}\rangle$ will then find all the qubits in the $|0\rangle$-state if the implemented function was constant or return at least one qubit in the $|1\rangle$-state for the case of a balanced function. Hence, thanks to the parallelism naturally implemented in quantum registers and the interference introduced by the Hadamard gates, only one query, implemented through the operator U_f, is necessary.

The analysis presented by Collins et al. highlighted another important aspect of the refined Deutsch-Jozsa algorithm. From Figure 6-1, it can be seen that the core of the algorithm lies in the operator U_f which has the form:

$$U_f = \begin{bmatrix} (-1)^{f(0)} & 0 & 0 & 0 \\ 0 & (-1)^{f(1)} & 0 & 0 \\ 0 & 0 & \ldots & 0 \\ 0 & 0 & 0 & (-1)^{f(2^n-1)} \end{bmatrix} \tag{6.2}$$

However, expression (6.2) describes U_f in an abstract form, which does not say how to implement this operator, for example, during a proper computation on a quantum computer. To do that, it would first be necessary to find a decomposition of U_f into the, typically, one- and two-qubit gates implementable by the chosen technology. In this context, Collins et al. showed that only when implementing the refined Deutsch-Jozsa algorithm for, at least, the three-qubit case, two-qubit gates would be necessary to solve it and that entanglement would be introduced in the quantum register. Moreover, even when implementing this algorithm for $n \geq 3$, not all balanced functions would require two-qubit interactions. Hence, when exploiting the refined Deutsch-Jozsa algorithm to show experimentally that a physical system correctly operates as a quantum computer, firstly, at least the three-qubit case would have to be considered. Secondly, out of all possible balanced functions $f_n(x)$, only those that require two-qubit interactions should be selected. These facts give important guidelines for the choice of the system to be analysed. The aim of the work here presented is to produce circuits suitable for the experimental implementation of a quantum computer based on the SFG model. However, as seen in Chapter 3, experimental quantum computation imposes exceptional technological challenges which result in systems of a few qubits (typically 2-4, with some exceptions) being implemented for experimental demonstrations. It was therefore decided to develop circuits for the 3-qubit case of the refined Deutsch-Jozsa algorithm in order to exploit it in its full complexity while keeping the technological challenges to the simplest possible level.

6.2 Designing circuits for a three-qubit refined Deutsch-Jozsa algorithm implemented in SFG technology

In the following sections, different implementations of circuits solving the refined Deutsch-Jozsa algorithm will be presented. As described in Chapters 2 and 5, finding a circuit for a quantum computation is equivalent to finding sequences of quantum gates which, once the corresponding gates' unitary operators are multiplied together, return the unitary operator describing the quantum computation one wants to implement. As can be seen from Figure 6-1, in the case of the refined Deutsch-Jozsa algorithm, the operator which needs to be decomposed into a quantum circuit is U_f, the unitary matrix describing the function implemented by the oracle.

In terms of the information stored in the system, one of the qubits is labelled q_2, the second q_1 and the third one q_0 with the state of the quantum register defined as $|q_2 q_1 q_0\rangle$. The three qubits are assumed to be located such that each of them can interact with the others. Finally, the function chosen for many of the examples here reported is f_{17}, where the function coding is the same used by Kim et al in [Kim00a] which labels each function with the hexadecimal value represented by its binary output string. As shown in (6.3), this function is associated to the binary string [0 0 0 1 0 1 1 1], which, on a decimal scale corresponds to 23 and to the hexadecimal value of 17. Expressions (6.3) also show the unitary operator *U₁₇* implementing this function.

$$\begin{array}{l} f(000)=0 \\ f(001)=0 \\ f(010)=0 \\ f(011)=1 \\ f(100)=0 \\ f(101)=1 \\ f(110)=1 \\ f(111)=1 \end{array} \quad U_{17}=\begin{bmatrix} (-1)^0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & (-1)^0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & (-1)^0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & (-1)^1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & (-1)^0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & (-1)^1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & (-1)^1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & (-1)^1 \end{bmatrix} \qquad (6.3)$$

This function is known to require 3 two-qubits gates for its implementation, see [Kim00a] for example, and represents, therefore, one of the functions with the maximum complexity. Below, the circuits which have been derived in the study here presented are reported.

6.2.1 Circuits obtained from an NMR experiment and adapted through local equivalence

The first circuit was obtained starting from a solution given in [Kim00a] for an NMR experiment and adapting it to the SFG case exploiting local equivalence. The circuit proposed by Kim et al. for the function f_{17} is the following:

$$U_{17}=R_{1z}(\pi)J_{01}\left(\frac{\pi}{2}\right)J_{12}\left(\frac{\pi}{2}\right)J_{02}\left(-\frac{\pi}{2}\right) \qquad (6.4)$$

where $R_{1z}(\vartheta)$ represents a rotation of qubit 1 around the z-axis in the Bloch sphere notation while $J_{ij}(\vartheta)$ is a two-qubit interaction between qubit *i* and *j* in an NMR system, described mathematically by:

$$J_{ij}(\vartheta)=e^{-i\frac{\vartheta}{2}Z_i\otimes Z_j} \qquad (6.5)$$

The final circuit is then obtained by inserting the gate sequence given in equation (6.4) in the circuit shown in Figure 6-1. The output state of this circuit can be evaluated by feeding the total gate sequence into the quantum logic simulator described in the previous chapter, obtaining:

$$|\psi_{outNMR}\rangle = \frac{1}{\sqrt{8}}\begin{bmatrix} 0 \\ 1+i \\ 1+i \\ 0 \\ 1+i \\ 0 \\ 0 \\ -(1+i) \end{bmatrix} \qquad (6.6)$$

The circuit given in expression (6.4) is transformed into one compatible with SFG technology by observing two things: First, the gates $J_{ij}\left(\pm\frac{\pi}{2}\right)$ are locally equivalent to the controlled-phase gate through the transformation[Nie03]:

$$\begin{aligned} J_{ij}\left(-\frac{\pi}{2}\right) &= P\left(\frac{\pi}{4}\right) CP_{ij} R_{zi}\left(-\frac{\pi}{2}\right) R_{zj}\left(-\frac{\pi}{2}\right) \\ J_{ij}\left(\frac{\pi}{2}\right) &= P\left(-\frac{\pi}{4}\right) CP_{ij} R_{zi}\left(\frac{\pi}{2}\right) R_{zj}\left(\frac{\pi}{2}\right) \end{aligned} \qquad (6.7)$$

where $P(\vartheta)=e^{-i\vartheta}$ is a constant phase shift while CP_{ij} is a controlled-phase gate between qubit *i* and *j*. Secondly, and as shown, for example, in [Rod04], the controlled-phase gate can be obtained via the SFG model. Hence, by replacing the $J_{ij}\left(\pm\frac{\pi}{2}\right)$ gates with controlled phase gates as shown in expression (6.7) and then by finding SFG gates which approximate controlled-phase gates, it is possible to implement the circuit given in (6.4) with an SFG-compatible approach.

The entangling space of the SFG gate was therefore searched for good approximations of controlled-phase gates considering values of *M* and *N* between 1 and 2500. Out of a set of about 25 equivalent candidates, the following three gates were selected due to the variability in their *f*-value (equation (4.3), Chapter 4) in order to mimic the variation in gate parameters which would be present in an experimental implementation of an SFG quantum computer because of the random distribution of qubits and control particles:

-SFG gate 1: *M*=1595, *N*=2137, *f*=5.348
-SFG gate 2: *M*=1584, *N*=2177, *f*=4.5 (6.8)
-SFG gate 3: *M*=815, *N*=904, *f*=18.89

All three gates approximate the controlled-phase gate with an average fidelity *AF* (equation (5.34), Chapter 5) >0.999 and were chosen as the entangling gates to be used for implementing the circuit given in (6.4). As shown schematically in Figure 6-2, SFG gate 1 was assumed to act between qubit 0 and 1, SFG gate 2 between qubit 1 and 2, SFG gate 3 on qubits 0 and 2.

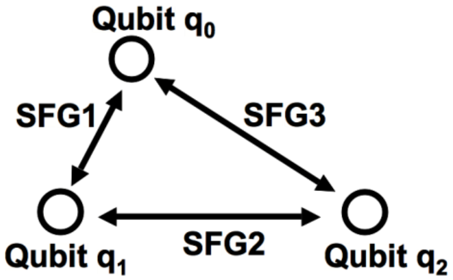


Figure 6-2: Distribution of SFG gates on a three-qubit SFG chip

In the first experimental implementations of an SFG chip such as the one schematised in Figure 6-2, it will be unlikely to have such a convenient distribution of the entangling gates. It has here been assumed that the *f*-values of the entangling gates can be imposed on the circuit. In reality, these values will be the outcome of a chip characterisation procedure[Sto08] after the random distribution of qubits and control particles and should, therefore, be considered as given data which might not correspond to the optimum value necessary to implement, for example, the above listed controlled-phase gates. Nevertheless, the ideal circuits produced through this assumption are a fundamental step which enable the comparison of different design strategies, as will be shown in Sections 6.2.2.2-6.2.2.4.

Hence, the operator U_{17} implemented with SFG technology has then been obtained by combining the gates presented in (6.8) with expressions (6.7) and (6.4). The corresponding complete circuit implementing the whole refined Deutsch-Jozsa algorithm is shown in Figure 6-3.

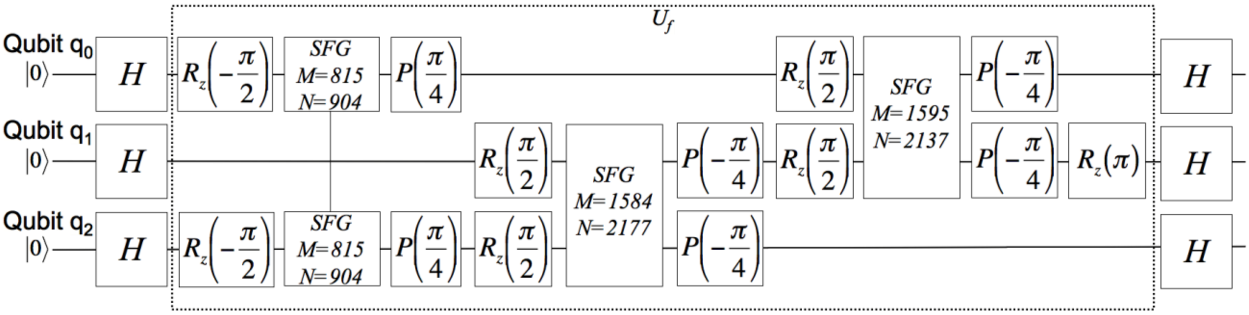


Figure 6-3: Total circuit for a three-qubit refined Deutsch-Jozsa algorithm with SFG gates accurately modelling controlled-phase gates

Simulating the circuit given in Figure 6-3 with the quantum logic simulator returns the final output state:

$$|\psi_{outSFG}\rangle = \frac{1}{\sqrt{8}}\begin{bmatrix} -0.01+i0.01 \\ 1+i \\ 1+i0.99 \\ 0.01-i0.01 \\ 0.99+i1.01 \\ -0.02+i0.02 \\ 0 \\ -(1+i) \end{bmatrix} \tag{6.9}$$

This output state can be compared with the ideal one obtained with the equivalent NMR circuit using the output-state fidelity (expression (5.30), Chapter 5):

$$fidelity = |\langle\psi_{ideal}|\psi_{err}\rangle|^2 \tag{6.10}$$

where, in this case, $|\psi_{ideal}\rangle$ is the output evaluated with the NMR circuit while $|\psi_{err}\rangle$ is the one approximated through the SFG circuit. Using expression (6.10) on output states (6.9) and (6.6) returns:

$$|\langle\psi_{out-ideal}|\psi_{out-SFG}\rangle|^2 = 0.9998 \tag{6.11}$$

6.2.2 Circuits obtained through automated quantum circuit design based on a genetic programming approach

The circuit presented in the previous paragraphs has been obtained from a solution derived for an NMR quantum computer [Kim00a]. In that case, the gate sequence had been obtained using a generator expansion technique [Kim00b]. Instead, for the reasons given in the previous chapter, the automated quantum circuit design algorithm based on a genetic programming approach proposed by Williams and Gray[Wil99] was used for deriving the remaining circuits presented in this thesis.

The analysis starts with decompositions found for the case of circuits comprising ideal, technology-independent controlled-phase gates. Then, focusing again on the operator U_{17}, circuits for different configurations of SFG gates are presented.

6.2.2.1 Circuits comprising ideal and technology independent controlled-phase gates

The genetic programming algorithm has been run for all 35 possible balanced functions of the refined Deutsch-Jozsa algorithm. Since from the analysis of the results of Kim et al. in [Kim00a], the angles of the single-qubit rotations were expected to be multiples of a fraction of π, all angles in this part of the design process were limited to the ensemble $\{-\pi,-7\pi/8,...,+7\pi/8,+\pi\}$. For all functions, an exact solution, i.e. one characterised by $AF=1$, was obtained after few iterations of the genetic programming algorithm. The results are reported using the same hexadecimal codification of the functions and sorting according to the number of two-qubit gates in the circuit used by Kim et al.[Kim00a]. Table 6-1 summarises the circuits found with the genetic programming tool.

Table 6-1: **Quantum circuits for all 35 balanced functions obtained through genetic programming**

Function	Circuit
0 Two-qubit gates	
f_{0F}	$R_{2z}(\pi)$
f_{33}	$R_{1z}(\pi)$
f_{3C}	$R_{1z}(\pi)R_{2z}(\pi)$
f_{55}	$R_{0z}(\pi)$
f_{5A}	$R_{0z}(\pi)R_{2z}(\pi)$
f_{66}	$R_{0z}(\pi)R_{1z}(\pi)$
f_{69}	$R_{0z}(\pi)R_{1z}(\pi)R_{2z}(\pi)$
1 Two-qubit gate	
f_{1E}	$R_{2z}(\pi)CP_{01}$
f_{2D}	$R_{0z}(-\pi)CP_{01}R_{2z}(\pi)$
f_{36}	$CP_{02}R_{1z}(\pi)$
f_{39}	$CP_{02}R_{1z}(-\pi)R_{2z}(-\pi)$
f_{4B}	$CP_{01}R_{0z}(\pi)R_{2z}(-\pi)$
f_{56}	$CP_{12}R_{0z}(-\pi)$
f_{59}	$R_{2z}(\pi)CP_{12}R_{0z}(-\pi)$
f_{63}	$CP_{02}R_{0z}(\pi)R_{1z}(\pi)$
f_{65}	$R_{0z}(\pi)R_{1z}(-\pi)CP_{12}$

f_{6A}	$R_{0z}(-\pi)R_{1z}(\pi)R_{2z}(-\pi)CP_{12}$
f_{6C}	$R_{1z}(-\pi)CP_{12}R_{0z}(-\pi)R_{2z}(\pi)$
f_{78}	$R_{1z}(-\pi)R_{2z}(\pi)CP_{12}R_{1z}(-\pi)$
2 Two-qubit gates	
f_{1B}	$CP_{01}\ CP_{02}R_{2z}(\pi)$
f_{1D}	$R_{2z}(\pi)CP_{01}\ CP_{12}$
f_{27}	$CP_{01}\ CP_{02}\ R_{1z}(-\pi)$
f_{2E}	$R_{1z}(-\pi)R_{2z}(-\pi)CP_{12}\ CP_{01}$
f_{35}	$R_{1z}(\pi)CP_{02}\ CP_{12}$
f_{3A}	$CP_{12}\ R_{2z}(\pi)\ CP_{02}\ R_{1z}(\pi)$
f_{47}	$CP_{02}\ R_{1z}(\pi)CP_{12}$
f_{4E}	$R_{2z}(\pi)CP_{01}\ CP_{02}\ R_{0z}(-\pi)$
f_{53}	$CP_{02}\ R_{0z}(\pi)CP_{12}$
f_{5C}	$CP_{12}\ R_{0z}(-\pi)\ CP_{02}\ R_{2z}(-\pi)$
f_{72}	$R_{1z}(-\pi)CP_{01}\ R_{0z}(-\pi)CP_{02}$
f_{74}	$R_{1z}(-\pi)CP_{01}\ CP_{12}\ R_{0z}(\pi)$
3 Two-qubit gates	
f_{17}	$CP_{01}\ CP_{02}\ CP_{12}$
f_{2B}	$CP_{12}\ CP_{02}\ CP_{01}\ R_{1z}(-\pi)R_{2z}(\pi)$
f_{4D}	$CP_{02}\ R_{2z}(\pi)CP_{01}CP_{12}R_{0z}(\pi)$
f_{71}	$CP_{12}\ R_{1z}(-\pi)CP_{01}CP_{02}R_{0z}(\pi)$

Comparing Table 6-1 and the results presented in [Kim00a], it can be seen that the circuits obtained with the two different methods require the same number of two-qubit gates. In terms of single-qubit gates, the same length of circuits has been found for all functions belonging to the group requiring 0 two-qubit gates. For the remaining functions, it was found that 2D, 39, 63, 59, 65, D8, AC, CA, 27, 47, 53, 1D, 35,17 designed with the genetic programming algorithm required 1 single-qubit gate fewer, functions 36 and 56 two fewer, while function 4D required one single-qubit gate more. However, these differences might not be caused by the different methods used for the decomposition, they could also have been induced by the different entangling gates used. The gates used by Kim et al. are based on NMR technology and are locally equivalent to the controlled-phase gate, but not identical to it.

It is also important to compare the circuit obtained with the genetic programming algorithm and ideal controlled-phase gates for the case of function f_{17} (Table 6-1) with the first solution presented in this chapter in Figure 6-3. In that case, the circuit had been obtained using the local equivalence of the controlled-phase gate with the gates available from the NMR implementation used by Kim et al.[Kim00a]. Specifically, controlled-phase gates and single qubit operations were used to replicate each of the two-qubit gates used in the circuit presented by Kim et al. Instead, using the genetic programming approach, we were able to decompose the circuit directly with controlled-phase gates, without using these to replicate other two-qubit gates and avoiding, therefore, the use of extra single-qubit gates.

6.2.2.2 Circuits exploiting SFG gates approximating controlled-phase gates

Aiming at investigating the resources needed for the demonstration of a quantum computer based on SFG gates, the next step was to understand how the circuits presented above may change when the controlled-phase gates used in the decomposition process were not ideal, but approximated via the SFG model. The operator U_{17} was chosen again for the decomposition process. Since at this stage the controlled-phase gates were not ideal but approximated, the angles in the single-qubit rotations were allowed to vary continuously between $-\pi$ and $+\pi$ assuming that fixed multiples of $\pi/8$ may not be optimal anymore. Using the same gates given in expression (6.8), the following circuit was obtained:

$$U_{17app1} = R_{z0}(-0.001)SFG1\,R_{z0}(0.008)SFG3\,R_{z1}(-0.011)R_{z2}(0.019)SFG2 \qquad (6.12)$$

U_{17app1} given in expression (6.12) approximates the ideal transformation U_{17} with an average fidelity AF=0.999978. The design algorithm was stopped after about 300 rounds when no appreciable increase of the average fidelity could be observed. In terms of output-state fidelity (expression (6.10)), the new circuit obtained a fidelity value of 0.99998 compared to the 0.9998 of the circuit shown in Figure 6-3. Hence, the three extra R_z rotations present in equation (6.12) compared to the shorter circuit shown in Table 6-1 partly compensate for the non-ideal controlled-phase gates generated by the SFG two-qubit interaction.

6.2.2.3 Circuits exploiting SFG gates implementing fast controlled-phase gates

Although the circuit given in expression (6.12) simulates the U_{17} operator with very high precision, it is also important to analyse the corresponding computation time. Using the expressions given in (4.3) of Chapter 4 and assuming the same magnetic field term used in some recent studies of the SFG gate dynamics of 0.136meV[Ker07], yields values of the gate operation times T_i between, approximately, 80ns and 160ns for the SFG gates used in Sections 6.2.1 and 6.2.2.2. The aim is to reduce these gate operation times, and, therefore, the total computational time, so that the chances for impairments such as decoherence, deteriorating the state of the qubits, are minimized. To achieve this, the entangling space of SFG gates was searched for fast gates modelling controlled-phase gates whilst accepting a lower precision compared to the set presented in (6.8).

For a static magnetic field term of 0.136meV, following SFG gates which approximate controlled-phase gates with *AF*>0.99 and a gate operation time T_i<10ns, were found:

-SFG1=SFG(124,142), *J*=51.93GHz, T_1=2.63ns
-SFG2=SFG(137,156), *J*=54.37GHz, T_2=2.77ns (6.13)
-SFG3=SFG(143,162), *J*=56.77GHz, T_3=2.77ns

With this set of gates, the use of the genetic programming tool helped identifying the circuit:

$$U_{17app2} = SFG2\, R_{z1}(0.038)\, SFG1\, R_{z0}(0.059)\, SFG3\, R_{z2}(0.183) \qquad (6.14)$$

which approximates the U_{17} operator with an average fidelity of *AF*=0.9888 and the final output-state of the total refined Deutsch-Jozsa algorithm circuit with a fidelity of 0.987. Again, compared to the shortest possible circuit which can be obtained with ideal controlled-phase gates, the circuit described in expression (6.14) uses three more single-qubit rotations. Without these, the average fidelity of the circuit implementing U_{17} is 0.979, confirming that through the genetic programming tool it is possible to find single-qubit rotations which compensate, in part, the non-ideal approximation of controlled-phase gates obtained via the SFG gates.

Summarising, at the expense of a lower average fidelity, the solution given in (6.14) approximates U_{17} with SFG gates more than ~30 times faster compared to those used in (6.12). Specifically, the reduction of a factor 30 in the gate operation time comes at the expenses of a loss of only ~1.3% in the output state fidelity.

6.2.2.4 Circuits exploiting fast SFG gates

In the results presented up to this point, the focus has been set on circuits based on controlled-phase gates and how they could be efficiently implemented through SFG gates. However, as shown in Chapter 4, there is a variety of entangling gates different from the controlled-phase gate which can be produced within the SFG model. Any of these entangling gates, together with single-qubit operations, forms a universal set of gates and is therefore sufficient for designing any unitary operator. Hence, in this section, it is explored whether it is possible to design faster circuits than those presented above by exploiting arbitrary entangling gates with the shortest possible gate operation time instead of approximations of controlled-phase gates. It is assumed to have a three-qubit system with three random values of *J*, one for each gate. Physically, this is equivalent to the situation of a semiconductor substrate hosting a random distribution of qubits and control atoms with their corresponding *J* values given, for example, by a characterisation process[Sto08]. Assuming a static magnetic field term *B* equal to 0.136meV, the value of *f*=*B/J* is calculated for each *J* and a procedure based on a continued fraction algorithm[Sav06,Ker07] is used for finding the fastest possible SFG gates which can be obtained with these parameters. The procedure used in [Ker07] aims at finding fast entangling gates typically used in literature, such as the C-NOT gate, for example, at the expense of having some residual entanglement between the control particle and the qubits at the end of the gate protocol. Here, the focus is on ideal SFG gates, in which qubits and control atoms are left unentangled at the end of a two-qubit operation, at the expense of having slower gates compared to the ones presented in [Ker07]. Similar parameters to those in [Ker07] are used, assuming for the first gate J_1=61.175GHz. J_2 and J_3 were then arbitrarily chosen at, respectively, 5GHz and 10GHz distance from J_1. The continued fraction algorithm returned following gate parameters:

-SFG1=SFG(73,82), J_1=61.175GHz, T_1=1.308ns, c_1=1.22, c_2=1.22, c_3=0.094
-SFG2=SFG(79,88), J_2=66.175GHz, T_2=1.3055ns, c_1=1.311, c_2=1.311, c_3=0.0003 (6.15)
-SFG3=SFG(85,94), J_3=71.175GHz, T_3=1.3031ns, c_1=1.754, c_2=1.387, c_3=0.0071

where the c_i parameters are the ones defined by Zhang et al. in [Zha03], and discussed in Chapter 4, which describe the location of the gates in the a^+ chamber. By analysing the c_i parameters given in expression (6.15), it can be seen that the three entangling gates are different from each other and from the controlled-phase gate which has c_i parameters [π/2,0,0][Zha03]. The genetic programming tool was run with these gates for different maximal allowed lengths of the circuit TQmax. However, in a compromise between precision of the circuit and length, TQmax=20 was the maximum length considered, obtaining as best result the complete circuit shown in Figure 6-4 which also includes the 3 Hadamard gates (represented through the compact notation $H^{\otimes 3}$) used at the beginning and end of the Deutsch-Jozsa algorithm. In Figure 6-4, some two-qubit gates are characterised by having a gate operation time $T=N_{rep}\cdot T_i$. This notation is used to describe sequences in which the genetic programming tool returned N_{rep} repetitions of the same two-qubit gate. Remembering that SFG gates are applied through the transmission of an exciting pulse and of a de-exciting pulse after a time T_i, N_{rep} repetitions of such a gate are equivalent to separating the exciting and de-exciting pulse by a time $N_{rep}\cdot T_i$ and experimentally correspond to the application of a single two-qubit gate. Hence, in the circuit given in Figure 6-4, the implementation of U_{17} requires 9 effective two-qubit gates and 14 single-qubit operations, achieving an average fidelity AF=0.9343, while the output state of the total circuit approximates the ideal one with a fidelity equal to 0.9677.

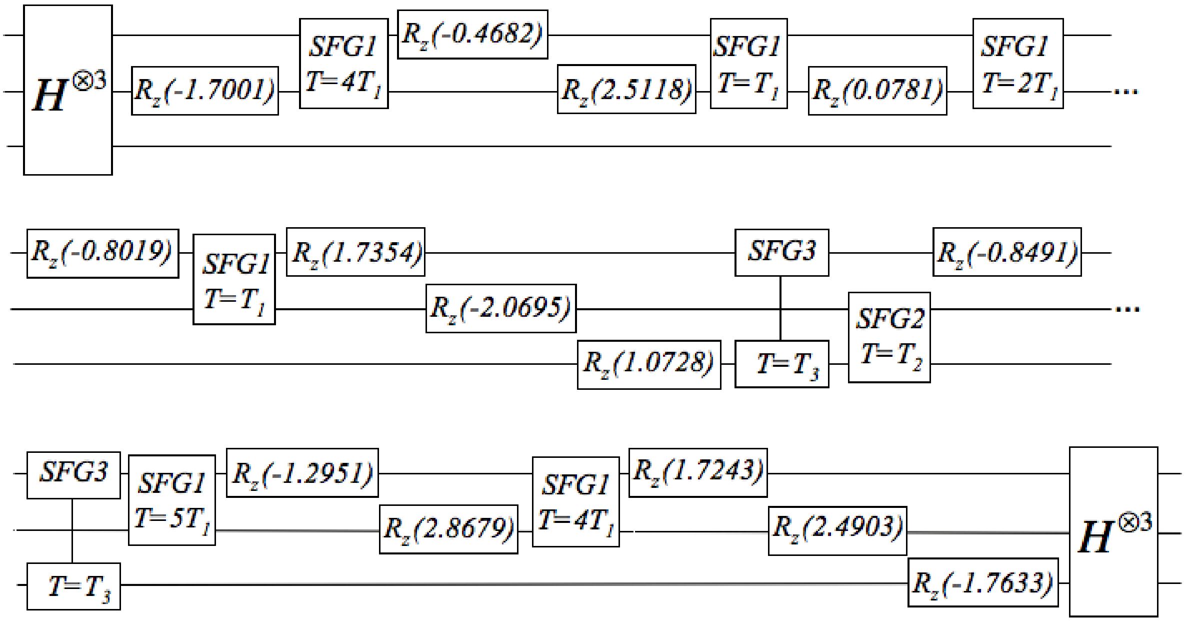


Figure 6-4: Refined Deutsch-Jozsa algorithm circuit obtained with arbitrary entangling gates

Comparing the circuit given in Figure 6-4 to the ones obtained using approximations of controlled-phase gates it can be seen that, when using arbitrary entangling gates, it was only possible to obtain a circuit which used more than three times the number of gates required when using SFG based controlled-phase gates. In terms of computational time, the circuit given in Figure 6-4 requires approximately $20\cdot T_i$ of time dedicated to two-qubit interactions (the sum of all the gate operation times for two-qubit gates) whereas the circuits exploiting controlled-phase gates only required approximately $3T_i$. Hence, despite the shorter computational time of the arbitrary entangling gates, the final circuit obtained using these gates had a longer total computational time and, moreover, achieved a lower average fidelity. The reason for this could be the following. As shown in [Col98] and reviewed at the beginning of the chapter, all operators implementing balanced functions for a refined Deutsch-Jozsa algorithm are diagonal, with the diagonal comprising a balanced distribution of ‘1s’ and ‘-1s’. The controlled-phase gate and the $R_{zi}(\theta)$ rotation (which was the only single-qubit operation appearing in the circuits based on controlled-phase gates, i.e. Sections 6.2.2.1-6.2.2.3) are also diagonal operators and their multiplication returns a diagonal operator. Hence, when using controlled-phase gates to implement another diagonal operator, the genetic programming tool simply needs to identify the sequence of controlled-phase gates and $R_{zi}(\theta)$ rotations which produces the required balanced distribution of ‘1s’ and ‘-1s’ corresponding to the chosen function. Conversely, as can be seen from expressions (4.2) in Chapter 4, an arbitrary SFG gate has two off-diagonal elements which, once multiplied with single-qubit operators, fill off-diagonal terms of the total function operator. The design process, when using arbitrary entangling gates, has to introduce the desired sequence of ‘1s’ and‘-1s’ on the diagonal and, at the same time, cancel out off-diagonal terms. Hence, although arbitrary entangling SFG gates and single-qubit operations form a universal set of gates, their structure may make the implementation of diagonal operators less efficient compared to using gates such as the controlled-phase gate. When implementing a three-qubit refined Deutsch-Jozsa algorithm, it appears, therefore, more efficient to choose the SFG parameters such that the corresponding entangling gates approximate controlled-phase gates, which can be done with the methods demonstrated by Kerridge et al [Ker07].

Between all the presented circuits, accepting a compromise between precision of the final circuit and computational time, the circuit given in expression (6.14) which implements the refined Deutsch-Jozsa algorithm with an output-state fidelity of 0.987 through SFG gates approximating controlled-phase gates with ~2.7ns gate operation time, appears to be the most suitable solution for the experimental implementation of a quantum computational system based on the SFG model.

6.2.3 Potential system errors

All the presented circuits have been derived under the assumption of perfect control of the quantum computational system. During an experiment, this is not going to be the case and it is therefore important to analyze how the performance of a circuit may change due to perturbations affecting the system. Here, the problem is addressed focusing on the circuit given in expression (6.14).

One source of errors in the above described circuit will come from the presence of decoherence. As mentioned in Section 5.1.4.2, it is important to distinguish the impact which decoherence will have on the qubits as opposed to the one which will be experienced by the control particles. In terms of decoherence on the qubits, studies have shown that for systems similar to the ones envisaged by the SFG model of qubits carried by the electron spin of donors in a solid-state substrate, the dominant effect is a T_2 process which leads to a loss of the phase coherence of the qubits and that it is reasonable to expect dephasing times in excess of milliseconds[Tyr03]. Considering the compactness of the circuit and its short gate operation times, this should not prevent the implementation of the algorithm. In terms of the control particles, perturbations will be introduced by the unwanted decay of the excited state of the controls through a relaxation event. This would lead to shorter interaction times between the qubits compared to the desired ones and, therefore, to the implementation of perturbed two-qubit gates as well as leakage of quantum information from the qubits to the controls. Unfortunately, there are few results which have been presented in literature on the lifetimes of the excited states of defect-substrate systems which may be compatible with the control particle scheme proposed within the SFG model. To date, to the best of our knowledge, the most important results for the SFG proposal are represented by the measurement of the lifetimes of the excited state of phosphorus atoms in a silicon substrate which have been presented in [Vin08] with reported lifetimes of the order of ~200ps. This lifetime is shorter than the gate operation times of the two-qubit gates used in the circuit shown in (6.14), which means that the use of phosphorus atoms as control particles in a silicon substrate would not be a convenient solution for implementing the presented circuit. However, other possibilities are being analysed such as, for example, double donors like Se^+ and Mg^+ in silicon or phosphorus atoms in a diamond crystal[Sto08], and it will be important to study whether longer lifetimes may be achieved through these systems.

There are three other causes of perturbation which can be identified in the analyzed systems and which may manifest themselves in an experimental scenario. In the above presented analysis, perfect knowledge and control over the parameters which regulate the SFG gate dynamics, i.e. T_i, J and B, has been assumed. In an experiment, none of these parameters would be known with such precision and, as can be seen from expressions (4.3a) in Chapter 4, their value directly influences the two-qubit quantum gates produced within the SFG model. Perturbations of these values will therefore reflect on the final outcome of the computation. The problem is best illustrated by analyzing the impact of timing-jitter which may affect the pulse interleave time T_i. Due to, for example, noise in the circuits used for the generation of the optical pulses or, as will be discussed in the next section, imperfections in the devices used for signal generation, the actual value of the pulse interleave time generated by such a system, may be perturbed to a value $T_{ierr}=T_i+\delta T_{tj}$, where δT_{tj} represents an uncontrollable timing-jitter component. The impact of this perturbation is twofold. Firstly, as mentioned above, the perturbed pulse interleave time T_{ierr} will produce a perturbed quantum transformation and, therefore, deteriorate the quantum information processed in the computation. Secondly, as demonstrated in [Rod04], SFG gates generated through parameters obtained using the equations given in (4.4) in Chapter 4 have negligible residual entanglement between qubits and control particles and avoid, therefore, the leakage of quantum information from the quantum register to the controls. Hence, the perturbed pulse-interleave time T_{ierr} will not only deteriorate the processed quantum information through a perturbed quantum transformation but also cause a loss of the quantum information stored in the qubits. Similar arguments hold for fluctuations on J and B. In case of J, uncertainties in the knowledge of its value may derive from the precision of the chip characterization procedure used to measure its value[Sto08]. For B, perturbations may be introduced by the inhomogeneity of the magnetic field along the quantum register. To quantify the impact of these uncertainties on the final outcome of the computation, the performance of the circuit given in (6.14) was evaluated under the influence of increasing perturbation. The focus has been here set on T_i and J since, from the equations given in (4.4), Chapter 4, it can be expected that perturbations introduced by B will be of the same order of magnitude of the ones introduced by J due to their proportionality relation through f. A Gaussian distribution was assumed for both the fluctuations[Obe99,Niw02]: on δT_{tj} (fluctuations on T_i) and δJ (fluctuations on J), with zero mean and standard deviation σ_T and σ_J, respectively. The output state fidelity of the circuit shown in (6.14) has then been evaluated when perturbing every SFG gate in the circuit with different random values of fluctuations and gradually increasing the standard deviation of the perturbation. For each value of standard deviation, the results were averaged over 1000 perturbed circuits[Niw02]. The results of the output state fidelity, averaged over the 1000 runs, are shown in Figure 6-5(a) and (b) for standard deviations σ_T and $\sigma_{J,}$ respectively, between 0 and 1.2ps for fluctuations on T_i and between 0 and 27MHz for fluctuations on J. As reported in Section 6.2.2.3, the circuit described by expression (6.14) achieved an output state fidelity of 0.987 with unperturbed gate parameters. In the presence of perturbations (Figure 6-5) the degradation in the average output state fidelity is below 1% if the fluctuations on T_i and on J have standard deviations of less than, respectively, ~0.3ps and ~7.7MHz while a degradation larger than 10% could be observed for standard deviations larger than, respectively, ~1ps and ~25.1MHz.

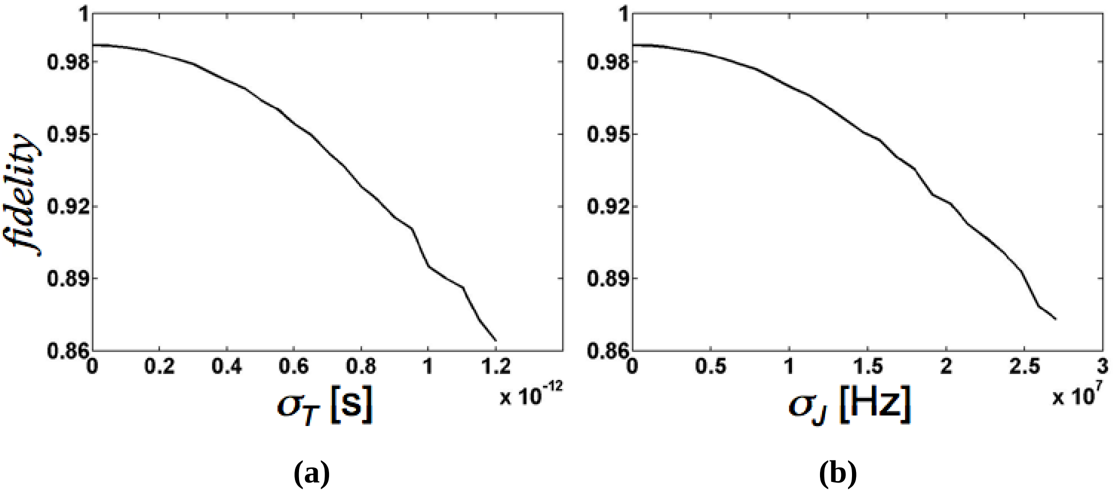


Figure 6-5: Output state fidelity degradation as a function of the standard deviation of the perturbation for:(a) T_i and (b) J.

6.3 The optical control signal and its generation

As discussed above, a convenient circuit for the experimental demonstration of a quantum computational system based on the SFG model is the one shown in expression (6.14) which requires three different SFG gates, one for each pair of qubits, to be applied to the quantum register. In an SFG quantum computer this corresponds to the excitation and de-excitation of the corresponding control particle for each qubit pair. Supposing that ω_1, ω_2 and ω_3 are, respectively, the excitation frequencies of the control particles corresponding to SFG1, SFG2 and SFG3 in (6.14), to implement that circuit one would have to transmit, first, two optical pulses centred on ω_3 and separated by a pulse interleave time T_3, then transmit the two pulses separated by T_1 and centred on ω_1 and, finally, the last pair of pulses separated by T_2 and centred on ω_2. Also, some buffer space ΔT_i between each pair of pulses will be necessary in order to leave time for single-qubit operations to be applied to the chip. Figure 6-6 shows schematically the optical signal necessary to control the two-qubit interactions when implementing the circuit given in expression (6.14).

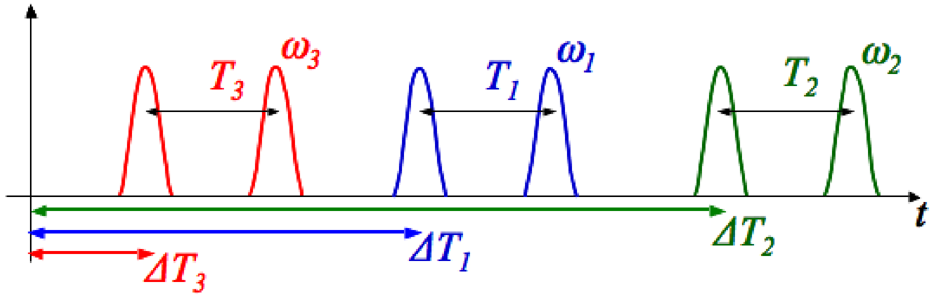


Figure 6-6: Schematic representation of the optical control signal necessary to implement the circuit given in expression (6.14).

Also, when implementing a two-qubit interaction with an SFG two-qubit gate it is necessary that the excited control electron couples with all qubit spin-states carrying the information. As described in [Rod04], this is achieved by exploiting exciting and de-exciting pulses with sufficiently broad spectrum which translates to pulsewidths of the order of the picosecond. Hence, the optical control signal of the two-qubit interactions in an SFG quantum computer is a multi-wavelength sequence of picosecond pulses.

6.3.1 Generating the optical signal

The excitation frequency of control particles in SFG two-qubit gates mainly depends on the materials used in the system. Recently, for example, the lifetime of the excited state of phosphorus control particles in silicon has been measured experimentally [Vinh08]. In that case, the excitation energy was 34.1meV, corresponding to a wavelength of 36.36μm, a part of the optical spectrum in which convenient lasers are not available. However, systems are currently being investigated in which excitation frequencies fall in a more accessible wavelength range. Examples are double donors like Se^+ as control particles in silicon or phosphorus impurities in diamond[Sto08], both of which are characterised by excitation wavelengths around 2.2-2.3μm[Ber89,Laz08].

To date, there are a number of different solutions for generating picosecond pulses in that range of wavelengths. Both Kivisto et al. [Kiv07] and Chan et al.[Cha08], for example, have recently demonstrated systems based on the Raman-induced soliton selffrequency-shift able to produce pulses with a pulsewidth ~0.1ps at a wavelength of ~2.15μm. While such systems would be extremely convenient in terms of compactness and simplicity, they do not seem able yet to fully cover the wavelength requirements of Se^+ in silicon and phosphorus in diamond.

The systems which seem most flexible in terms of wavelength range and signal characteristics are those based on optical parametric conversion, where the second-order nonlinearity of a crystal is used to transform a high-energy pump radiation into two lower frequency signals, called *signal* and *idler* (see, for example, [Lau74]). As reviewed, for example, in [Pet01], the frequency of signal and idler is typically tuned by changing the propagation angle of the fields inside the crystal or by exploiting a tunable pump signal as well as, in periodically poled crystals (i.e. ferroelectric crystals with a periodic reversal of the domains), by changing their temperature or their modulation period.

A large number of systems based on optical parametric conversion and able to cover the 2.2-2.3μm wavelength range have been demonstrated experimentally, with a wide variety in the output signal characteristics, such as the degree of tunability, the output pulsewidths and pulse energies, reported from set-up to set-up. Examples range from one of the early systems demonstrated by Lauberau et al. [Lau74], which reported the generation of picosecond pulses tunable between 1.4-4μm with a pulse energy of up to 10 μJ, to the one presented by Butterworth et al.[But96], which generated pulses with a pulse width of around 2.6ps, tunable between 1.67-2.806μm and pulse energies in the order of 1nJ, up to the recent experiment by Brida et al.[Bri08], which demonstrated femtosecond pulses, tunable between 2-5μm with an output energy of up to 2μJ. Systems with compatible signal output characteristics are also commercially available. The Spectra-Physics OPA-800CP [Spe09], for example, is a system based on an optical parametric amplifier (OPA), pumped by a Ti:sapphire system, which produces picosecond pulses at a repetition rate of 1 kHz, tunable between 1.1-3μm and output pulse energies of up to tens of μJ. Considering that, to the best of our knowledge, there is no OPA system able to tune the frequency quickly enough to produce the signal required by the SFG protocol, three different OPAs would be necessary to generate a signal similar to the one shown in Figure 6-6. However, observing that in the experiments presented by Vinh et al. in [Vin08] pulse energies of up to 16.7nJ were typically exploited which only represent a fraction of the power which can be obtained from a system like the OPA-800CP, a single pump system would be sufficient to feed three different OPAs. The optical control signal for a three-qubit refined Deutsch-Jozsa algorithm performed on an SFG quantum computer may, therefore, be generated with a system schematically shown in Figure 6-7.

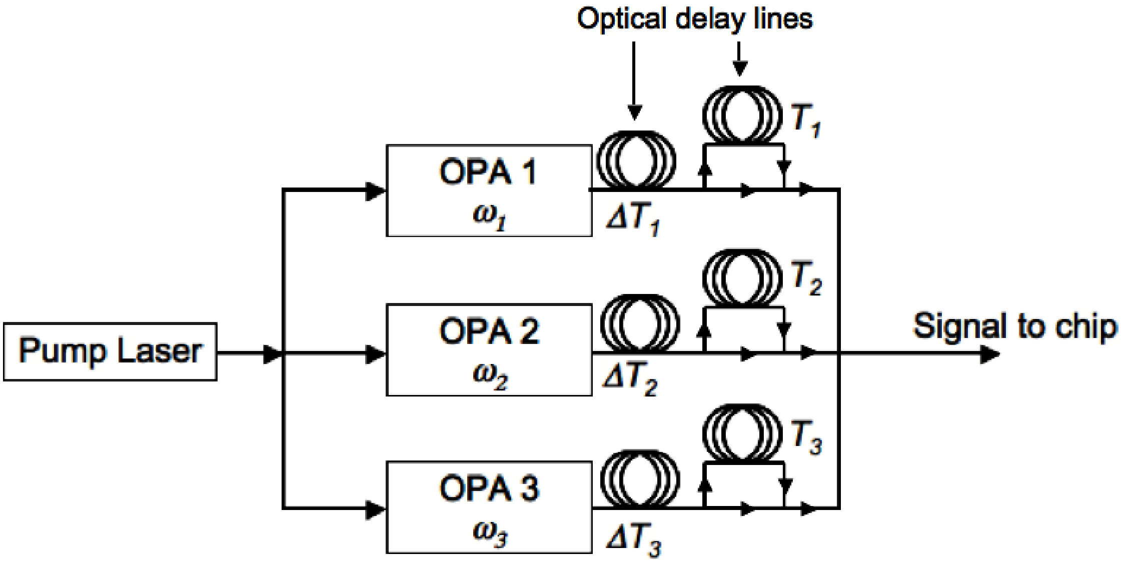


Figure 6-7: Three-wavelength OPA system for generating picosecond optical pulses in the 2.2-2.3μm wavelength range.

In Figure 6-7, a pulse leaving the pump system is split into three identical pulses which are subsequently transmitted into the three OPA systems. Each OPA is tuned on the required frequency ω_i. Inside the OPAs, the three pump pulses are then transformed into three synchronised pulses respectively centred at frequency ω_1, ω_2 and ω_3. A first set of optical delay lines is used to introduce the delay ΔT_i between the different frequencies (i.e. between the different two-qubit gates) and is also responsible for introducing buffer space for implementing single-qubit operations when necessary. Typically reported values of single-qubit gate operation times in experiments concerning donor electron spin qubits are in the range of some tens of nanoseconds[Kop06,Mor08], corresponding to, approximately, a few meters of fibre. Then, splitters and a second set of optical delay lines are used for generating, at each frequency, the excitation and de-excitation pulses with the correct pulse interleave time T_i. Here, the delay is expected to be around 2.7ns (expressions (6.13)), corresponding to less than 1m of fibre. In terms of precision, remembering from the error analysis presented in Section 6.2.3 that the standard deviation of timing jitter should be kept below 0.3ps for achieving a decay in the average fidelity below 1%, commercially available optical delay line kits (for example, the Optical Delay Line Kit produced by Newport [New09]) specify delay sensitivities below 10fs and can, therefore, be used to fine-tune the pulse interleave times T_i without introducing excessive timing jitter. The three signals are finally recombined and transmitted to the chip. Summarising, a system as the one shown in Figure 6-7 should be able to produce the sequence of multiwavelength picosecond pulses, tunable between 2.2-2.3μm necessary for the experimental demonstration of a three-qubit refined Deutsch-Jozsa algorithm on an SFG quantum computation system.

However, OPA systems as the ones described above are bulky. While such a solution would provide an important tool for testing the main features of SFG quantum computation on few-qubit systems, it would be impractical in terms of system integration and scalability to larger set-ups. Considering as well that other possible double donors which could be used as control particles in a silicon system may require longer excitation wavelengths than the ones discussed above (e.g. ~5μm for Mg^+[Sto08, Thi94]), suggests that the development of more compact and integrable optical systems for the generation of picosecond pulses at wavelengths larger than 2.2μm would be extremely beneficial for an optically controlled quantum computation system such as the one proposed within the SFG model.

In this context, although to date, to the best of our knowledge, no system conveniently matching the signal characteristic described above has been reported, quantum cascade lasers[Fai94] could represent a valuable resource for increasing the scalability of the SFG quantum computation proposal. As described, for example, in [Cap02], quantum cascade lasers are compact sources able to produce optical radiation in the mid-infrared part of the spectrum and for which picosecond pulse generation has been achieved[Pai00]. Being based on semiconductor technology, they may be effectively integrable in a solid-state quantum register as the one analyzed in this thesis.

6.4 Summary

The design of optically-controlled quantum logic circuits suitable for the experimental demonstration of a three-qubit quantum computer based on SFG gates has been analysed. The algorithm chosen for this study is the refined Deutsch-Jozsa algorithm because, even on few-qubit quantum registers, it allows one to demonstrate the three main features of quantum computation: parallelism, interference and entanglement. The aim of this analysis was to identify strategies for the design of the shortest possible circuits in order to reduce the probability of errors accumulating along the computation path. This led to a set of four different design procedures. While the first one was based on adapting to the SFG technology a circuit developed for an NMR quantum computer, the remaining three were based on a genetic programming approach for quantum circuit design and differed in the two-qubit entangling gates used during the design process. Out of the 4 solutions, an optimal one was identified as a result of trade-off between circuit precision and computation time. This circuit was obtained using the genetic programming approach in conjunction with SFG gates approximating fast controlled-phase gates. The tolerance of this circuit towards perturbations was then tested by analysing the decrease in output state fidelity for increasing fluctuations of some two-qubit gate parameters. It was found that, for fluctuations on T_i and J characterised by standard deviations of up to, respectively, 0.3ps and 7.7MHz, the resulting decrease in output state fidelity was below 1%.

Finally, observing that in an SFG quantum computer ultimately quantum logic circuits are implemented through the transmission of optical pulses, possible approaches to the generation of the optical control signal were addressed. Focusing on some potential control particle candidates, an initial design of a system based on optical parametric amplification was proposed which could generate the multiwavelength, picosecond pulse sequences centred around the 2.2μm wavelength range, required to control the quantum logic circuits proposed. It was also observed, however, that the development of more compact and integrable picosecond optical systems in the near mid-infrared and mid-infrared part of the spectrum would be highly desirable for SFG based quantum computation.

6.5 References

[Ber89] K. Bergman, G. Grossmann, H.G. Grimmeiss, M. Stavole, R.E. McMurray, “Applicability of the deformation-potential approximation to deep donors in silicon”, *Physical Review B,*vol. 39, pp.1104-1119, (1989)

[Bri08] D. Brida, M. Marangoni, C. Manzoni, S. De Silvestri, G. Cerullo, “Sub-two-cycle light pulses at 1.6 µm from an optical parametric amplifier”, *Optics Letters*, vol.33, pp.2901-2903, (2008)

[But96] S.D.Butterworth, V.Pruneri, D.C.Hanna, “Optical parametric oscillation in periodically poled lithium niobate based on continuous-wave synchronous pumping at 1.047µm”, *Optics Letters*, vol.21, pp. 1345-1347, (1996)

[Cap02] F.Capasso, R.Paiella, R.Martini, R.Colombelli, C.Gmachl, T.L.Meyers et al., “Quantum cascade lasers: Ultrahigh-speed operation, optical wireless communication, narrow linewidth, and far-infrared emission”, *IEEE Journal of Quantum Electronics*, vol.38, pp.511-532, (2002)

[Cha08] M.C. Chan, S.H. Chia, T.M. Liu, T.H. Tsai, M.C. Ho, A.A. Ivanov et al., “1.2-2.2-mu m tunable Raman soliton source based on a Cr : fosterite laser and a photonic-crystal fiber”, *IEEE Photonics Technology Letters*, vol.20, pp. 900-902, (2008)

[Chu98] I.L. Chuang, L.M.K. Vandersypen, X.L. Zhou, D.W. Leung, S.Lloyd, “Experimental realization of a quantum algorithm”, *Nature*, vol.393, pp.143-146, (1998)

[Cle98] R.Cleve, A. Ekert, C.Macchiavello, M.Mosca, “Quantum algorithms revisited”, *Proceedings of the Royal Society of London Series A-Mathematical Physical and Engineering Sciences*, vol.454, pp.339-354, (1998)

[Col98] D. Collins, K.W. Kim, W.C. Holton, “Deutsch-Jozsa algorithm as a test for quantum computation”, *Physical Review A*, vol. 58, pp. R1633-R1636, (1998)

[Deu92] D.Deutsch, R.Jozsa, “Rapid solution of problems by quantum computation”, *Proceedings of the Royal Society of London Series A-Mathematical Physical and Engineering Sciences*, vol.439, pp.553-558, (1992) [Fai94] J.Faist, F.Capasso, D.L.Sivco, C.Sirtori, A.L.Hutchinson, A.Y.Cho, “Quantum Cascade Laser”, *Science*, Vol.264, pp.553-556, (1994)

[Jon98] J.A. Jones, M.Mosca, “Implementation of a quantum algorithm on a nuclear magnetic resonance quantum computer”, *Journal of Chemical Physics*, vol.109, pp1648-1653, (1998)

[Ker07] A. Kerridge, A.H. Harker, A.M. Stoneham, “Electron dynamics in quantum gate operation”,*Journal of Physics: Condensed Matter*, vol. 19, article number 282201, (2007)

[Kim00a] J. Kim, J.S. Lee, S. Lee, C. Cheong, “Implementation of the refined Deutsch-Jozsa algorithm on a three-bit NMR quantum computer”, *Physical Review A*, vol. 62, Article number 022312, (2000)

[Kim00b] J.Kim, J.S.Lee, S.Lee: “Implementing unitary operators in quantum computation”, *Physical Review A*, Vol.61, Article number 032312, (2000)

[Kiv07] S. Kivisto, T. Hakulinen, M. Guina, O.G. Okhotnikov, “Tunable Raman soliton source using mode-locked Tm-Ho fiber laser”, *IEEE Photonics Technology Letters*, vol. 19, pp.934-936, (2007)

[Kop06] F.H.L.Koppens, C.Buizert et al.: “Driven coherent oscillations of a single electron in a quantum dot”, *Nature*, Vol.442, p.766-771, (2006)

[Lau74] A. Laubereau, L.Greiter, W.Kaiser, “Intense tunable picosecond pulses in infrared”, *Applied Physics Letters*, vol.25, pp.87-89, (1974)

[Laz08] A.Lazea, V.Mortet, J.D’Haen, P. Geithner, J. Ristein, M.D’Olieslaeger, K. Haenen, ‘Growth of polycrystalline phosphorus-doped CVD diamond layers”, *Chemical Physics Letters*, vol. 454, pp.310-313, (2008)

[Mor08] J.J.L.Morton, A.M.Tyryshkin, R.M.Brown, S.Shankar, B.W.Lovett et al.: “Solid-state quantum memory using the ^{31}P nuclear spin”, *Nature*, vol.445, pp.1058-1088, (2008)

[New09] http://www.newport.com/store/genproduct.aspx?id=396220&
lang=1033&Section=Spec

[Nie03] M.A. Nielsen, I.L.Chunag, *Quantum Computation and Quantum Information*, Cambridge: Cambridge University Press, 2003

[Niw02] J.Niwa, K.Matsumoto, H.Imai, “General-purpose parallel simulator for quantum computing”, *Physical Review A*, vol.66, Article number 062317, (2000) [Obe99] K.M. Obenland, A.M.Despain, "Simulating the effect of decoherence and inaccuracies on a quantum computer", *in Quantum Computing and Quantum Communications, Book Series: Lecture Notes in Computer Science*, vol. 1509, pp.447-459, (1999)

[Pai00] R.Paiella, F.Capasso, C.Gmachl, D.L.Sivco, J.N.Baillargeon, A.L.Hutchinson et al., "Self-mode-locking of quantum cascade lasers with giant ultrafast optical nonlinearities", *Science*, vol.290, pp.1739-1742, (2000)

[Pet01] V. Petrov, F. Rotemund, F. Noack, "Generation of high-power femtosecond light pulses at 1 kHz in the mid-infrared spectral range between 3 and 12 mu mby second-order nonlinear processes in optical crystals", *Journal of Optics A-Pure and Applied Optics*, vol.3, pp.R1-R19, (2001)

[Rod04] R. Rodriquez, A.J. Fisher, P.T.Greenland, A.M. Stoneham, "Avoiding entanglement loss when two-qubit quantum gates are controlled by electronic excitation", *Journal of Physics: Condensed Matter*, vol.16, pp.2757-2772, (2004)

[Sav06] S. Savory, "Disentangling the control electron in a two qubit solid state quantum gate", *Journal of Physics: Condensed Matter*, vol. 18, pp. S777-S782, (2006)

[Spe09] http://www.newport.com/store/genproduct.aspx?id=368110&
lang=1033&Section=Spec

[Sto08] A.M. Stoneham, "The quantum in your materials world", *Materials Today*, vol.11, pp. 32-36, (2008)

[Thi94] A. Thilderkvist, M. Kleverman, H.G. Grimmeiss, "Interstitial magnesium double donor in silicon", *Physical Review B*, vol.49, pp.16338-16348, (1994)

[Tyr03] A.M.Tyryshkin, S.A.Lyon et al.: "Electron spin relaxation times of phosphorus donors in silicon", *Physical Review B*, vol.68, Article number 193207, (2003)

[Vin08] N.Q. Vinh, P.T.Greenland, K. Litvinenko, B. Redlich, A.F.G van der Meer, S.A. Lynch et al.: "Silicon as a model ion trap: Time domain measurements of donor Rydberg states", *Proceedings of the National*

Academy of Sciences of the United States of America, vol.105, pp.10649-10653, (2008)

[Wil99] C.P. Williams, A.G. Gray, “Automated design of quantum circuits”, *in Quantum Computing and Quantum Communications, Book Series: Lecture Notes in Computer Science*, vol. 1509, pp.113-125, (1999)

[Zha03] J.Zhang, S.Sastry, K.B.Whaley, “Geometric theory of nonlocal two-qubit operatrions”, *Physical Review A*, vol.67, Article number 042313, (2003)

Chapter 7 Conclusions and future work

7.1 Conclusions

This thesis described research carried out which aimed to investigate and propose quantum logic circuits suitable for the experimental demonstration of a few-qubit quantum computational system based on SFG two-qubit gates. Specifically, the focus was set on identifying quantum circuits which, in an experimental demonstration of a few-qubit SFG quantum computation prototype, would prove that the prototype under analysis correctly implements the main features of quantum computation, i.e. entanglement, interference and parallelism. During the research the following questions were addressed. Firstly, it was necessary to analyse typical system parameters of state-of-the-art experimental quantum computation to identify the possibilities and limitations of a prototype quantum computer experimental demonstration. In particular, it was necessary to quantify the number of qubits and types of test-algorithms commonly used for such experiments. Then, the quantum gates feasible within the SFG model had to be identified in order to establish which entangling gates can be used for the design of quantum logic circuits based on this proposal. Another unknown was the quantum circuit design strategy for the derivation and implementation of circuits using SFG two-qubit gates. A requirement of these strategies was the design of short quantum circuits to minimise the probability of errors accumulating during computation. Finally, because SFG quantum logic gates are controlled optically, optical systems for the generation of the required control signal for the SFG-based circuits had to be proposed.

Part of these questions were initially addressed in the context of numerous existing proposals and experimental demonstrations which helped highlight the main challenges of quantum computation and the main results of this analysis are reported in Chapter 3. Through this review it emerged that, to date, although one of the most sensational experimental demonstrations of quantum computation was a factorization algorithm with seven qubits, experiments on smaller registers (e.g. 2 to 4 qubits, with few exceptions of larger systems) are commonly reported. In terms of algorithms, the Deutsch-Jozsa algorithm, and particularly its refined version, is often used in the experimental demonstration of quantum computational systems because it allows one to implement parallelism, interference and entanglement, as long as it is performed on a quantum register of at least three qubits. Hence, with the aim of the development of quantum circuits which would prove that an SFG quantum computation prototype effectively operates as a quantum computer, whilst minimising the technological challenges, the three-qubit refined Deutsch-Jozsa algorithm was selected in this work as the test problem to be developed.

The subsequent steps of the research focused therefore on the design of quantum logic circuits implementing a three-qubit refined Deutsch-Jozsa algorithm compatible with the SFG model. To do that, first, the SFG two-qubit entangling gate was analysed. Moreover, because only a discrete ensemble of pulse interleave times T leaves the control particle unentangled from the qubits at the end of the SFG gate protocol, it was important to study which entangling gates can be produced out of this ensemble. This analysis was performed using a geometrical method which allows one to visualise entangling gates in a three dimensional space called Weyl chamber and the results of this study are reported in Chapter 4. It was shown that the SFG model generates a variety of entangling gates which uniformly covers the surface of the Weyl chamber. Solutions of SFG gates close to points indicating specific quantum gates typically used in the literature correspond to approximations of these gates obtained through the SFG model. The analysis of the entangling gates produced through SFG gates led to two observations:

1) The distribution of SFG gates on the Weyl chamber suggests that, when using the SFG model to approximate, for example, the controlled-phase gate, a certain flexibility in the choice of the gate parameters is available. Given the requirement for circuits with the shortest computational time, this flexibility can then be exploited in the choice of the gates used in the design process to trade-off their approximation accuracy against the gate operation time.

2) The uniform distribution of solutions of SFG gates over the Weyl chamber also suggests that two-qubit gates different from the controlled-phase gate can be considered when designing circuits for the experimental demonstration of an SFG quantum computation system.

The significance of these two observations was in establishing the guidelines used to select the SFG-based quantum gates for the design of the quantum circuits.

Two numerical tools were developed in the course of the work aimed at solving the quantum circuit design problem. These tools are presented in Chapter 5. The first one addresses the issue of the analysis of quantum logic circuits based on SFG gates. In the literature, algorithms for analyzing the performance of quantum logic circuits had been proposed and these can be used for studying, for example, how the performance of a quantum circuit may decay under the influence of perturbations or to compare how well different circuits implement the same function. However, these algorithms focus on the state of the qubits by studying their evolution under the effect of the quantum gates comprised in the circuit to be analysed. In SFG quantum computation, the transformation which is applied to the qubits also depends on the state of the control particles. Unwanted and unpredictable changes in the state of a control particle lead to perturbations in the corresponding SFG gates and, therefore, to errors in the computation. Hence, in the analysis of a quantum circuit based on SFG gates, it is necessary to follow the evolution of the qubits as well as that of the control particles to get a more complete description of the computation. Existing algorithms for the analysis of quantum logic circuits were, therefore, further developed in order to obtain a quantum logic simulator specifically tailored for the SFG model. This was achieved by including the states of the control particles in the state vector used for analysing the evolution of the computation as well as introducing, a part from the gates commonly used in the literature such as the controlled-phase or the C-NOT gate, the transformations corresponding to SFG gates in the library of the simulator. The significance of this work was is in generating a numerical tool for the study of how the specific dynamics of SFG gates can influence the result of a quantum computation.

The second numerical tool addressed, rather than the issue of the analysis of quantum logic circuits, the actual problem of their design. Previously demonstrated quantum circuit design techniques used for deriving circuits for three-qubit refined Deutsch-Jozsa algorithms were difficult to adapt to the case of SFG computation due to the presence of the control particle in the dynamics of the gate. Also, these techniques typically require the C-NOT gate or the controlled-phase gate to always be part of the quantum gates library used in the design process. However, the analysis of the entangling characteristic of SFG gates had revealed that, apart from approximations of controlled-phase gates, other entangling gates were available. A design technique which would not put any constraints on the quantum gates library was therefore required, in order to analyze whether circuits with improved performance may have been obtained using gates different from approximations of controlled-phase gates. This flexibility was found in the automated quantum circuit design algorithm based on a genetic programming approach proposed by Williams and Gray[1] which was therefore implemented for SFG gates. In the perspective of deriving quantum circuits suitable for the experimental demonstration of an SFG quantum computation prototype, the development of this second numerical tool provided a fundamental instrument for studying how the topology of circuits may change depending on the choice of the SFG gates used in the design process and, therefore, allowed the testing of different strategies for obtaining convenient circuits.

Quantum circuits implementing a three-qubit refined Deutsch-Jozsa algorithm with SFG gates were considered in Chapter 6. Different design routes were tested and compared with the aim of identifying efficient circuits characterised by short computational times in order to reduce the chances of errors accumulating during computation. The first circuit was obtained adapting to SFG technology a circuit designed for an NMR quantum computer exploiting the local equivalence of entangling gates produced in NMR with the controlled-phase gate and identifying SFG gates which produced controlled-phase gates with an average fidelity >0.999. The resulting circuit generated an output state fidelity of 0.9998. The remaining circuits were designed using the genetic programming approach described in Chapter 5, testing different sets of gates and analysing the impact on the circuits' performance of the type of SFG gates used. The first circuit obtained with the genetic programming approach used the same SFG gates as the solution obtained from the NMR circuit. However, despite being based on the same SFG gates, a more compact and improved circuit (output state fidelity ~0.99998) was obtained. The increase in the output state fidelity is the result of optimised single qubit operations, identified through the genetic programming approach, which compensate for part of the non-ideal approximation of the controlled-phase gates of the SFG model. Unfortunately, the SFG gates used in this circuit modelled high-accuracy controlled-phase gates at the price of long gate operation times, in excess of 80ns. Hence, a new circuit was developed, using the genetic programming algorithm, which exploited SFG gates with gate computation times T<10ns approximating controlled-phase gates with an average fidelity >0.99, instead of 0.999. The resulting circuit performed less well, with an output state fidelity of 0.987 (which

[1] *Quantum Computing and Quantum Communications, Book Series: Lecture Notes in Computer Science*, vol. 1509, pp.113-125, (1999)

corresponds to a reduction of only ~1.3% compared to the previous solution) with the advantage, however, of using quantum gates with gate operation time *T*~2.7ns, i.e. more than a factor of 30 faster. Finally, the last circuit discussed exploited a different approach in terms of the choice of the entangling gates. Remembering that any set of gates comprising entangling gates and single qubit operations is universal for quantum computation, the last solution was obtained, still using the genetic programming approach, exploiting arbitrary entangling gates, meaning gates which do not resemble the ones typically used in literature such as the controlled-phase, the C-NOT or the $\sqrt{SWAP}$ gate, as long as they were characterised by gate operation times shorter than ~2.7 ns (i.e. less than the gates used in earlier examples). However, despite the shorter gate operation time, the circuit obtained with these gates only achieved an output state fidelity of 0.9677 requiring a time dedicated to two-qubit interactions longer, approximately, by a factor 3 compared to the solution comprising gates with *T*~2.7ns. The reason for this is probably linked to the different structure of the arbitrary entangling gates. The core operator of the Deutch-Jozsa algorithm is diagonal. So are the controlled-phase gate and the $R_z(\theta)$ rotation, which was the single qubit operation required for implementing the core operator of the Deutsch-Jozsa algorithm. Combinations of controlled-phase gates and $R_z(\theta)$ rotations lead to operators which are, again, diagonal. Hence, when using these gates, the genetic programming algorithm only needs to find the correct sequence of gates which generates the desired combination of '1s' and '-1s' on the diagonal of the core operator corresponding to the function one wants to implement. Instead, arbitrary entangling SFG gates have two off-diagonal elements which require the genetic programming algorithm to find the sequence of desired '1s' and '-1s' on the diagonal while at the same time cancelling out off-diagonal elements. Although arbitrary entangling SFG gates together with single-qubit operations form a universal set of quantum gates, their non-diagonal structure makes the design procedure for a diagonal operator such as the core transformation of the refined Deutsch-Jozsa algorithm less efficient. Conversely, it is more convenient to implement the refined Deutsch-Jozsa algorithm by forcing the SFG gates to approximate controlled-phase gates which can be done through the methods demonstrated by Kerridge et al.[2]. Out of all the presented circuits, accepting a compromise between precision of the final circuit and computational time, the circuit

[2] *Journal of Physics: Condensed Matter*, vol.19, article number 282201, Jul.2007

implementing the refined Deutsch-Jozsa algorithm with an output-state fidelity of 0.987 through SFG gates approximating controlled-phase gates with ~2.7ns gate operation time was chosen as the optimal solution for the experimental implementation of a quantum computational system based on the SFG model. The tolerance towards perturbations of this circuit was, therefore, assessed by evaluating its performance under increasing fluctuations of the SFG gates parameters. It was shown, that, if the optical control signal used for implementing the two qubit SFG gates is affected by timing-jitter with standard deviation <0.3ps then the decay in performance of the quantum circuit is <1%. A similar reduction can be expected if the values of the *J* parameters describing the strength of the interaction between qubits and controls in SFG gates (which, in an experimental implementation of an SFG quantum computer, would be obtained through a chip characterisation procedure) are known up to an off-set characterised by standard deviation of <7.7MHz.

Finally, optical systems able to produce the necessary control signal for the analysed quantum circuits are described and discussed in the last part of Chapter 6. Particularly, the initial design of a system developed with the assumption of two potential control particle schemes, i.e. the double-donor selenium in silicon and phosphorus in diamond, which both require excitation wavelengths between 2.2-2.3μm, is presented. This system is based on three independent, singly pumped optical parametric amplifiers and would be able to produce the three-wavelength, picosecond pulse sequences required for the implementation of a three-qubit refined Deutsch-Jozsa algorithm with SFG two-qubit gates. However, while such a system based on optical parametric amplification represents a valid tool for performing an experimental demonstration of a small prototype of SFG quantum computer, more compact and integrable sources would be necessary in the long-term for realizing scalable systems.

The quantum logic circuits and results presented in this thesis provide important guidelines for enabling the future experimental demonstration of a quantum computation system based on SFG gates. Firstly, the topology of the SFG-based circuits solving the three-qubit Deutsch-Jozsa algorithm allows one to define timescales with respect to the tolerance towards decoherence of the qubits and control particles. As summarised above, the fastest circuit exploited SFG gates with gate computation time of ~2.7ns. Hence, control particles used for implementing the given circuit need to be characterised by relaxation times of the excited state larger than 2.7ns. Similarly, considering that single qubit operations with electron spin qubits are typically characterised by gate computation times of some tens of nanoseconds, it is reasonable to assume that the implementation of the whole circuit will require ~100ns, which defines a lower bound on the tolerance towards decoherence of the qubits. In addition, the studies on the impact of fluctuations of the SFG gate parameters on the performance of the circuit help to quantify the precision required for the optical signal generation and the chip characterisation procedure or the order of magnitude of the decay in system performance which can be expected in case this precision cannot be met. Finally, the analysis of which optical systems could generate the optical signal necessary for the implementation of the proposed circuit identifies how this problem can be solved in the near future as well as describing its limitations in the context of systems comprising a larger number of qubits.

7.2 Future work

7.2.1 Quantum logic circuits based on fast SFG gates with residual entanglement between qubits and control particles

As discussed above, the refined Deutsch-Jozsa algorithm can be implemented more efficiently with SFG gates approximating controlled-phase gates as opposed to arbitrary entangling SFG gates. This result was here derived on an assumption of a convenient distribution of qubits and control particles, implying a flexible choice of the gate parameters when implementing controlled-phase gates. However, in an experimental scenario, the value of J would typically be a result of a measurement procedure for characterising SFG gates after the random distribution of the qubits and control particles. Hence, the values of J for different SFG gates will depend on the random distribution of the particles rather than being a parameter which can be flexibly imposed on a given system. Given a specific value of J, obtained from a characterisation procedure of a random distribution of particles on the computation chip, it will often not be possible to directly implement a controlled-phase gate through the integers M and N and the equations given in (4.3), Chapter 4. However, the techniques demonstrated by Kerridge et al. (reference given in footnote 2 on page 165) allow one to identify gate parameters able to produce accurate approximations of gates equivalent to controlled-phase gates with short gate computational time, at the expense of some finite but small residual entanglement present between qubits and control atoms at the end of the gate protocol. These gates can be produced without imposing the value of J on the system. Hence, the next step in this work would be to repeat the design procedures for the three-qubit refined Deutsch-Jozsa algorithm again considering approximations of controlled-phase gates this time obtained through the methods presented by Kerridge et al.. In this case, it will be important to analyse the impact of the residual entanglement between qubits and controls on the final output state of the computation.

7.2.2 Estimating the impact of decoherence

All the circuits presented in this work have been derived and analysed without considering the effect of decoherence. In reality, although decoherence on the qubits should not prevent the implementation of the presented circuits due to their compactness and the promising tolerance towards decoherence demonstrated by donor electron spin qubits in the solid-state, it will, nevertheless, affect their performance and it is important to quantify this perturbation. Further, because SFG gates make use of a control particle for mediating the interaction between two qubits, it will be important to assess the effect on the computation of decoherence affecting not only the qubits, but also the control atoms.

More specifically, the two impairments which are more likely to perturb computation are dephasing of the qubits and the unpredicted relaxation of a control particle from its excited states while it is mediating the interaction between two qubits. The first impairment leads to a gradual perturbation of the stored quantum information while the second leads to the implementation of an erroneous two-qubit interaction since the qubits interact for a shorter time compared to the ideal case. In order to assess the tolerance of the SFG model towards decoherence it will be important to simulate the performance of the presented circuits when affected by these two impairments.

This analysis could be performed as follows. First, the performance of the circuit solving the Deutsch-Jozsa algorithm is analysed assuming that the system is only affected by a dephasing mechanism perturbing the qubits. Starting from the ideal case, the performance of the circuit should be simulated for increasing amount of dephasing, i.e. running a series of simulations each characterised by a smaller T_2 value for the qubits. By evaluating the fidelity of the computation for each value of T_2 it will be possible to study how rapidly the performance of the circuit decreases when the qubits are perturbed by increasing decoherence. This study will allow to find an estimate of the minimum value T_{2min} which leads to an acceptable performance of the circuit and, consequently, to identify defects which, having a T_2 value larger than T_{2min} for a given substrate, would be suitable for implementing the proposed circuit in an experimental demonstration.

A similar analysis should then be implemented for analyzing the impact of relaxation affecting the control particles. In this case, the relevant parameter is the relaxation rate T_1. Again, by monitoring how quickly the performance of the circuit decreases for decreasing values of T_1, it will be possible to estimate a threshold value T_{1min}. This value could then be used to identify suitable control-particle candidates by studying which defects, in a given substrate, are characterised by a T_1 value larger than T_{1min}. The analysis on the impact of decoherence on the qubits and on the controls should be performed separately in order to understand how strongly each impairment perturbs the performance of the circuit. However, the estimates of T_{2min} and T_{1min}, could then be refined by analysing how the performance of the circuit changes under the combined effect of the two decoherence mechanisms.

Appendix List of acronyms

CP	Controlled-phase
C-NOT	Controlled-NOT
NMR	Nuclear magnetic resonance
OPA	Optical parametric amplifier
QPC	Quantum point contact
$\sqrt{SWAP}$	Root swap
SFG	Stoneham, Fisher, Greenland proposal for quantum computation
SQUID	Superconductive quantum interference device