الجمهوريةالجزائريةالديمقراطية الشعبية

Algerian Republic Democratic and Popular

وزارة التعليم العالى و البحث العلمى

Ministry of Higher Uducation and Scientific Research

جامعة حسيبة بن بوعلي

Hassiba Ben Bouali University of Chlef

Faculty of Electronics

Option : nanotechnologies

Thesis title

"Le transport de spin dans les nanomatériaux application les ordinateurs quantiques"

spin transport in nanomaterials application to quantum computers

Presented by :

- ***OUMSALEM Abdellah***

Jery members:

President:	**B.BELMADANI**	**Professeur (U.CHLEF)**
Promoter :	**M.BENAROUS**	**M. de conference (UHBC)**
Examiners:	**M.ALI BENAMARA**	**M.de conferences (UHBC)**
	M.LADREM	**Professeur (ENS.KOUBA)**
	M. HACHEMAN	**Dr. (USTHB)**

Promotion : 2007/2008

ACKNOWLEDGEMENTS

First and foremost, I would like to praise ALLAH the Almighty for every single gift He granted us and for His help and guidance all along our life and studies.

I would like to offer my sincere gratitude to my advisor, Dr BENAROUS Mohamed, for all his time, support and encouragement. Most importantly, I thank him for helping me to stay on the narrow path, and for not letting me stray into all the details of this subject, that I find so fascinating.

This thesis would not have been the same without the contributions of my respectful teachers, namely, Mr A. ALI BENAMARA, Mrs KOUADRI, Mr KERKOUD, Mr BELMILOUDI, and Mr TALEB. And the administration staff of the Electronics Department for their help. Friends have been of great encouragement in this process, for which we are so much thankful, DJAMAL, MAAMAR, MHAMEDI BOUZINA, FATIMAZOHRA, ALIA, KARIMA, RABIA, NADYA, SLIMAN, AHMED, ALI, YACIN and all the others.

Lastely, I would like to thank all my family who, during my studious life, gave me love and support to work and succeed

…………TO THE SOUL OF MY FATHER

…………………TO THE PERSON WHO MAKES ME A MAN (MY MOTHER)

……………………….TO ALL MY TEACHERS.

OUMSALEM ABDELLAH

Contents

Abbreviations

Figures, tables and symbols lists

General introduction…………………………………………………………………................. 1

Chapter I: Computers microarchitectures and the limits of semiconductors technology 5

I-1- Introduction……………………………………………………………………….. 5

I-2- Evolution of semiconductor technology……………………………………….... 5

I-3- Moore's law………………………………………………………………………... 6

I-4- Limits of technology scaling……………………………………………………… 8

I-5- Limits in frequency scaling…………………………………………………….. 10

I-5-1- Cycle time…………………………………………………………………… 10

I-6- Limits in low power……………………………………………………………….. 12

I-7- Computational integrity……………………………………………………….... 14

I-8- Future Directions in Microprocessor Systems…………………………………….. 15

I-8-1- Microprocessor today - microprocessor tomorrow………………………….. 16

I-8-2- Future directions in microarchitectures……………………………………… 17

I-8-2-1- Multithreading or multiprocessing……………………………….... 17

I-8-2-2- Simultaneous-multithreading (SMT)……………………………..... 18

I-8-2-3- Chip multiprocessor (CMP)………………………………………... 19

I-9- Networks on chips……………………………………………………………… 19

I-10- Conclusion…………………………………………………………………….. 20

Chapter II: Notions of quantum mechanics……………………………………………….. 21

II-1- Introduction……………………………………………………………………. 21

II-2- Basic concepts in quantum mechanics…………………………………………… 21

II-2-1- Wave-particle duality………………………………………………………. 21

II-2-2- Heisenberg's uncertainty principle…………………………………………. 22

II-2-3- Quantum superposition……………………………………………………... 24

II-2-4- Measurement theory………………………………………………………… 24

II-2-5- Entanglement……………………………………………………………….. 27

II-2-6- Entanglement and quantum indistinguishability…………………………… 29

II-2-7- Decoherence………………………………………………………………… 31

II-3- Linear Algebra and the Dirac notation………………………………………… 32

II-3-1- The Dirac notation and Hilbert spaces…………………………………… 32

II-3-2- Dual vectors………………………………………………………………… 33

II-3-3- Operators…………………………………………………………………… 35

II-3-4- The Spectral theorem…………………………………………………….. 38

II-3-5- Functions of operators……………………………………………………. 39

II-3-6- Tensor products…………………………………………………………... 40

II-3-7- The Schmidt decomposition theorem……………………………………. 42

II-3-8- Mixed states………………………………………………………………… 43

II-3-8-1- Mixed states……………………………………………………. 44

II-3-8-2- Mixed states and the Bloch sphere……………………………... 45

II-3-9- Time-evolution of a closed system………………………………………. 45

II-3-10- Composite systems………………………………………………………. 47

II-4-Qubits………………………………………………………………………………... 48

II-4-1- The state of a quantum system…………………………………………… 48

II-4-2- General quantum operations…………………………………………….. 52

II-4-3- The Hadamard transformation…………………………………………... 53

II-4-4- Single Qubit transformations…………………………………………….. 53

Chapter III: The use of spin in quantum computers……………………………………… 57

III-1- Introduction……………………………………………………………………… 57

III-2- Electron spin…………………………………………………………………….. 57

III-2-1- Toward the world of spin……………………………………………….. 57

III-2-2- Electron intrinsic angular momentum…………………………………... 58

III-2-3- Electron spin magnetic moment………………………………………… 58

III-2-3-1- Zeeman interaction and Zeeman effect…………………………... 59

III-2-3-2- The electron spin g-factor……………………………………… 61

III-2-4- Spin orbit interaction…………………………………………………….. 61

III-3- General concepts in spin-transport…………………………………………… 62

III-3-1- Basic transport mechanism in a magnetic device………………………. 62

III-3-2- Transport regimes……………………………………………………….. 64

III-3-2-1- Diffusive transport…………………………………………….. 64

III-3-2-2- Ballistic transport……………………………………………… 65

III-3-2-3- Tunneling………………………………………………………. 65

III-3-3- Crossover between different transport regimes……………………………. 65

III-3-3-1- Spin-transport at the atomic level………………………………… 66

III-3-3-2- Molecular spin transport………………………………………… 67

III-4- Transport theory: linear response……………………………………………….. 68

III-4-1- GMR "Giant magneto resistence"………………………………………. 68

III-4-2- TMR "Tunneling magneto resistence"………………………………….. 69

III-4-3- Spin-transport through carbon nanotubes and nanowires………………….. 70

III-5- Conception of quantum computers and DiVincenzo criteria…………………… 72

III-6- Experimental achievements…………………………………………………... 74

III-6-1- Single and coupled quantum dots……………………………………….. 74

III-6-2- Charge and spin control in quantum dots……………………………….. 77

III-6-3- Spin relaxation……………………………………………………….. 78

III-6-4- Optical interaction and optical readout of spins…………………………… 80

III-6-5- Spin initialization…………………………………………………….. 81

III-7- Proposals for quantum computing……………………………………………. 81

III-7-1- Quantum dot quantum computing………………………………………. 82

III-7-2- Quantum computing and the quantum Hall effect…………………………. 84

III-8- Obstacles to quantum dot quantum computing……………………………….. 85

III-8-1- Flying qubits and entanglement generation……………………………... 85

III-8-2- Gating error……………………………………………………………. 86

III-9- Future Goals……………………………………………………………….... 87

III-9-1- Detection of single-electron spin decoherence…………………………….. 87

III-9-2- Single-qubit rotations………………………………………………… 88

III-9-3- Two-Qubit gates……………………………………………………… 88

III-10- Conclusions……………………………………………………………….. 90

Chapter IV: Quantum computers……………………………………………………….. 92

IV-1- Introduction…………………………………………………………………… 92

IV-2-1- The way to the quantum computation……………………………………... 92

IV-2-2- Qubits and quantum parallelism………………………………………… 92

IV-2-3- Readout and probabilistic nature of quantum computers………………….. 93

IV-3- The Quantum circuit model…………………………………………………... 94

IV-3-1- Quantum gates………………………………………………………... 96

IV-3-1-1- Qubit gates……………………………………………………….. 96

IV-3-1-2- Controlled-U gates…………………………………………………….. 99
IV-3-2- Universal sets of quantum gates…………………………………………… 100
IV-3-3- Efficiency of approximating unitary transformations……………………... 103
IV-3-4- Implementing measurements with quantum circuits……………………… 105
IV-4- The programation process of a quantum computer……………………………... 107
IV-5- Elementary quantum algorithms…………………………………………………… 108
IV-5-1- Probabilistic versus quantum algorithms………………………………….. 108
IV-5-2- Deutsch algorithm…………………………………………………………. 113
IV-5-3- Deutsch-Jozsa algorithm…………………………………………………... 117
IV-6- Shor's factoring algorithm……………………………………………………… 121
IV-6-1- Exponential speed-up in Shor's factoring algorithm……………………… 121
IV-6-2- Classical part………………………………………………………………. 122
IV-6-3- Quantum Fourier transform……………………………………………….. 123
IV-6-4- Joining the pieces………………………………………………………….. 124
IV-7- Case of study "comparison between the linear 'classical' search algorithm and Grover's search algorithm"………………………………………………………………………. 126
IV-7-1- The classical search algorithm……………………………………………... 126
IV-7-2- Quantum mechanical algorithms…………………………………………... 128
IV-7-3- The problem……………………………………………………………….... 130
IV-7-4- Algorithm………………………………………………………………….. 130
IV-7-5- Explanation of the algorithm………………………………………………. 131
IV-7-6- How fast is it possible to find the desired element………………………… 134
IV-7-7- Implementation considerations……………………………………………. 135
IV-7-8- The programme of Grover's search algorithm in MATLAB programme…. 135
IV-7-9- Simulation of Grover algorithm with MATLAB programme: Example of 6nqubits……………………………………………………………………………………... 136
IV-8- Conclusion……………………………………………………………………….. 139
General conclusion………………………………………………………………………… 141
Appendix A 143
Appendix B 149
Appendix C 153
Bibliography... 157

ASIC: Application Specific Integrated Circuit

ILP : Instruction Level Parallelism

DSP: Digital signal processor

I/O: Input/Output

ISA: Instruction Set Architecture

PEs: Processing Elements

SMT: Simultaneous Multithreading

CMP: Chip Multiprocessor

RF: Radio Frequency

MEMS: Microelectromechanical System

SoC: System on a Chip

LEED: Low Energy Electron Difraction

RHEED: Reflection Hich Energy Electron Difraction

SIMS: Secondary ion mass Spectrometry

XPS: X-ray photo electron Spectrometry

UPS : Ultraviolet photoelectron Spectrometry

ArF: Argon Fluoride

UV: UltraViolet

KrF: Krypton Fluoride

FET: Field Effect Transistor

MOS: Metal Oxide Semiconductor

CMOS: Complementary Metal Oxide Semiconductor

MOSFET: Metal Oxide Semiconductor Field Effect Transistor

GMR: Giant Magneto Resistance

TMR: Tunneling Magneto Resistance

P: Parallel

AP: Anti Parallel

CPP: Current flowing Perpendicular to the Plane

CIP: Curent flowing In the Plane

LCAO: Linear Combination of Atomic Orbitals

RGF: Retarded Green Function

KKR: Korringa, Kohn and Rostoker

LKKR: Layer Korringa, Kohn and Rostoker

AF: Anti Ferromagnetic

FM: Feromagnetic

Qubit: Quantum bit

Cbit: Classical bit

QED: Quantum cavity Electrodynamics

NMR: Nuclear Magnetic Resonance

2DEG: Two Dimentional Electro Gas

QPC: Quantum Point Contact

hh: heavy hole

lh: light hole

ESR: Electron Spin Resonance

ODMR: Optical Detection of Magnetic Resonance

STIRAP: Stimulated Raman Adiabatic Passage

Figures list

Figure I-1: Trends in future size over time 6

Figure I-2: Evolution of transistor count of CPU/microprocessor and memory ICs 8

Figure I-3: Interconnect capacitances 9

Figure I-4: Wave pipelined vector multiplication 12

Figure I-5: The National Technology Roadmap For semiconductor: a- total transistor per chip b- On-chip local clock 15

Figure I-6: a- The Microprocessor today ***b-*** The Microprocessor tomorrow 17

Figure I-7: Parallelism profile for a parallel thread model 18

Figure I-8: How three different architectures partition issue slots: a- superscalar b- multithreaded superscalar c- SMT 18

Figure I-9: Chip multiprocessors 19

Figure II-1: A source of emits two qubits in an entagled state. Top: A two particle interferometer verification. Bottom: The principle in terms of one photon gates. 28

Figure II-2 : Three particles entanglement in a GHZ state. 29

Figure II-3: The state of a deterministic classical bit can be represented as one of two points, labelled ‘0’ and ‘1’. 50

Figure II-4: A probabilistic classical bit. Here the probabilities p_0 and p_1 of the bit being 0 and 1, respectively, are represented by the position of a point on the line segment between the points representing 0 and 1. 51

Figure II-5: State of a qubit on the Bloch sphere. 52

Figure II-6: The 50/50 beamspliter (top) and the corresponding diagram using the Hadamard transform (below). 54

Figure II-7: A Mach-Zehner interferometer (top) is a sequence of two Hadamard transformations (bottom). 55

Figure II-8 : Top : Mach-Zehnder interferometer including a phase shifter ϕ in one of the two beams. Bottom: The equivalent representation with Hadamard transformations and a phase shifter gate. 56

Figure III-1: Scheme of a spin-valve in the two resistance states: **a)** high resistance, **b)** low resistance. 62

Figure III-2: Magnetoresistance mechanism in a Co/Cu/Co spin valve (panel a) in the two spin current approximation. 64

Figure III-3: Magnetoresistance mechanism in a spin valve constructed with an atomic scaled spacer in the two spin current approximation. 67

Figure III-4: Schematic representation of a typical GMR experiment: ***a)*** Current In Plane (CIP) configuration, ***b)*** Current Perpendicular to the Planes (CPP) configuration. 69

Figure III-5: Cartoon showing the levels alignment in the magnetic point contact. The solid (dashed) line denotes a majority (minority) spin molecular state. ***a)*** symmetric case at zero bias, ***b)*** symmetric case at positive bias, and ***c)*** symmetric case at negative bias. 71

Figure III-6: Device (right) used to read-out the charge state of a quantum dot with a quantum point contact (QPC).	74
Figure III-7: ***(a)*** Scanning electron micrograph of a gated double dot structure with two adjacent quantum point contacts (QPCs). ***(b)*** Charge stability ("honeycomb") diagram of the double quantum dot.	75
Figure III-8: (a) SEM micrograph of an electrically gated double quantum dot structure with neighbouring QPC charge detectors. ***(b)*** Large-scale plot of the differential conductance dGS2/dV6 as a function of the voltages V2 and V6 applied to gates 2 and 6, respectively. In ***(c)*** and ***(d)***, GD and dGS2/dV6 are shown, respectively, as a function of V2 and V6 in the region close to the (1,0) to (0,1) transition.	76
Figure III-9: Different designs for etched structures of coupled quantum dots.	77
Figure III-10: Self-assembled InAs quantum dot structures. ***(a)*** AFM picture of dots grown at random locations. ***(b)*** Transmission electron microscope (TEM) cross-section of vertically stacked dots (indicated by arrows), ordered along the growth axis. ***(c)*** AFM picture of laterally ordered dots. This image was generated after prepatterning of the substrate. ***(d)*** Sketch of a three-dimensional lattice of dots that could be obtained by combining the growth methods of ***(b)*** and ***(c)***.	78
Figure III-11 : Quantum dot spin filter. ***(a)*** Only electrons in the state $\left|\downarrow\right\rangle$ are transported through the dot. ***(b)*** Only the spin ground state $\left|\uparrow\right\rangle$, can pass through the (empty) dot. In ***(c)*** and ***(d)***, the measured differential conductance dI/dV$_{SD}$ is shown for the cases ***(a)*** and ***(b)***, respectively, with tunnelling current I and source-drain voltage V$_{SD}$. In ***(e)***, we show a scheme of the theoretically predicted dI/dV$_{SD}$	79
Figure III-12: Experimental demonstration of a spin filter. The figures (a) and (b) show the focusing peak height as a function of the quantum dot gate voltage Vg.	81
Figure III-13: **a-** Two neighbouring electron spins confined to quantum dots **b-** An array of exchange-coupled quantum dots.	82
Figure III-14: Schematic diagram illustrating the Fermi contact hyperfine interaction.	85
Figure III-15: SEM picture and scheme of a structure to apply a local rf magnetic field to a quantum dot.	89
Figure IV-1: A quantum circuit.	94
Figure IV-2: A general (possibly irreversible) quantum operation or superoperator can be realized using a unitary operation by adding an ancilla and tracing out part of the output.	95
Figure IV-3: The measurement of the quantum state $\alpha_0\left|0\right\rangle+\alpha_1\left|1\right\rangle$ results in a quantum output $\left|b\right\rangle$ with probability $\|\alpha_b\|^2$ ($b \in \{0, 1\}$) together with a classical label 'b' indicating which outcome was obtained.	95
Figure IV-4: The NOT gate rotating the state $\left|0\right\rangle$ to the state $\left|1\right\rangle$.	96
Figure IV-5: The c-*U* gate.	99
Figure IV-6: Given a circuit *CU* implementing a unitary *U*.	100

Figure IV-7: Circuit implementing a Von Neumann measurement with respect to the basis $\{|\varphi_j\rangle\}$. 105

Figure IV-8: Another circuit implementing the Von Neumann measurement. 106

Figure IV-9**:** A circuit computing the parity of three qubits. 106

Figure IV-10: A circuit implementing a parity measurement. 107

Figure IV-11: A classical probabilistic computation acting on a register that can be in one of four states labelled 0, 1, 2, 3. 109

Figure IV-12**:** The classical probabilistic computation viewed in a quantum setting. 110

Figure IV-13: A fully quantum computation. 111

Figure IV-14: A quantum circuit exhibiting no quantum interference. 111

Figure IV-15: A quantum circuit exhibiting interference. 112

Figure IV-16: A circuit implementing the Deutsch algorithm. The measured value equals $f(0) \oplus f(1)$. 115

Figure IV-17: The circuit for Deutsch's algorithm with the $c-U_{f(x)}$ drawn instead of U_f. 117

Figure IV-18: A circuit for the Deutsch–Jozsa algorithm. 118

Figure V-19: The circuit of a discrete quantum Fourier transform on three qubits. 124

Figure IV-20: *Inversion about average* operation. 132

Figure IV-21: The *inversion about average* operation 133

Tables lists

Table I-1: Complexity of microchip and Moore's law 7

Table I-2: Capacity and speed increasing during the past period 8

Symbols

C: Complex number

P: Spin polarization

H: Hilber space

FO4 (Fan out Four): Unit used in computer measurement

L: Typical thickness of the layer

GCD: Greatest Common Divisor

λ_{emf}: elastic mean free path

l_ϕ: phase coherent length

l_{sf} : spin diffusion length

d_{exc}: exchange length

E: Energy of the system

Résumé

Utiliser le spin d’un électron comme « bit quantique » (formant ainsi un qubit) pour manipuler et stocker l’information quantique représente l’une des options les plus prometteuses de la nanotechnologie à envisager pour la réalisation d’un nanoprocesseur quantique. Le domaine qubit de spin a connu ces dernières années un essor prodigieux. Ainsi lorsqu’un électron est piégé dans une boîte quantique défini dans une nanostructure, on peut mesurer son spin, le manipuler et le faire interagir de manière cohérente avec un autre spin, définissant ainsi que le transport de spin. Le chaînon expérimental manquant pour compléter les opérations essentielles dans un ordinateur quantique est le transport cohérent d’un spin électronique. Il permettra d’envisager la réalisation d’ordinateurs quantiques.

Nous nous proposons dans ce projet d’étudier le phénomène de transport de spin électronique et de l’utilisé le dans la réalisation des ordinateurs quantiques, ainsi que dans l'implémentation des algorithmes quantiques.

Summary

During the past forty years astounding advances have been made in the manufacture of computers. The number of atoms needed to represent a bit in memory has been decreasing exponentially since 1950. Likewise the numbers of transistors per chip, clock speed, and energy dissipated per logical operation have all followed their own improving exponential trends. This rate of improvement cannot be sustained much longer; at the current rate in the year 2020 one bit of information will requite only one atom to represent it. The problem is that at that size the behaviour of a computer's components will be dominated by the principles of quantum physics.

In our work we talk about the exploitation of the spin transport in nanomaterials to realise quantum computer. In addition we simulate some of the quantum algorithm to show the differences between the classical computer and the quantum one.

A part from the computational power of a quantum computer there is a much more banal argument for incorporating quantum mechanics into computer science: Moore’s law and the limits of semiconductors technology (we will discuss Moore’s law, Computers microarchitectures and the limits of semiconductors technology in the first chapter of this thesis “Computers microarchitectures and the limits of semiconductors technology”). In 1965 Intel co-founder Gordon Moore observed an exponential growth in the number of transistors per square inch on integrated circuit and he predicted that this trend would continue. In fact, since then this density has doubled approximately every 18 months. If this trend continues then around the year 2020 the components of computers are at the atomic scale where quantum effects are dominant. We have thus to inevitably cope with these effects, and we can either try to eliminate them as long as this is possible and keep on doing classical computing or we can at some point try to make use of them and start doing quantum computing.

Quantum mechanics is one of the cornerstones of modern physics. It governs the behaviour and the properties of matter in a fundamental way, in particular on the microscopic scale of atoms and molecules. Hence, what we may call a classical computer, is itself following the rules of quantum mechanics. However, such devices are not quantum computers in the sense that all the inside information processing can perfectly be described within classical information theory. In fact, we do not need quantum mechanics in order to explain how the bits inside a classical computer evolve. The reason for this is that the architecture of classical computers does not make use of one of the most fundamental features of quantum mechanics, namely the possibility of superpositions. (We will discuss the superposition principal and other principals of quantum mechanics in the second chapter of this thesis “Notions of quantum mechanics”). Throughout the entire processing of any program on a classical computer, each of the involved bits takes on either the value zero or one. Quantum mechanics, however, would in addition allow superpositions of zeros and ones, that is, bits – now called qubits (quantum-bits) – which are somehow in the state zero and one at the same time. Computing devices which exploit this possibility, and with that all the essential features of quantum mechanics, are called quantum computers. Since they have an additional capability they are at least as powerful as classical computers: every problem that can be solved on a classical computer can be handled by a quantum computer just as well. The converse, however, is also true since the dynamics of quantum systems is governed by linear differential equations, which can in turn be solved (at least approximately) on a classical computer. Hence, classical and quantum computers could in principle emulate each other and quantum computers are thus no hypercomputers. So why quantum computing? And if there is any reason, why not just simulate these devices on a classical computer? To answer theses questions we must talk about the roles of quantum computers.

One reason for aiming at building quantum computers is that they will solve certain types of problems faster than any (present or future) classical computer – it seems that the border between easy and hard problems is different for quantum computers than it is for their classical counterparts. Here easy means that the time for solving the problem grows polynomially with the length of the input data (like for the problem of multiplying two numbers), whereas hard problems are those for which the required time grows exponentially. Prominent examples for hard problems are the travelling salesman problem, the graph isomorphism problem, and the problem of factoring a number into primes. For the latter it was, to the surprise of all, shown by Peter Shor in 1994 that it could efficiently be solved by a quantum computer in polynomial time. Hence, a problem which is hard for any classical computer becomes easy for quantum computers. Shor’s result gets even more brisance from the fact that the security of public key encryption, i.e., the security of home banking and any other information transfer via the internet, is heavily based on the fact that factoring is a hard problem.

One might think that the cost for the gained exponential speedup in quantum computers is an exponential increase of the required accuracy for all the involved operations. This would then be reminiscent of the drawback of analog computers. Fortunately, this is not the case and a constant accuracy is sufficient. However, achieving this “constant” is without doubt experimentally highly challenging.

Moreover, we know that nature provides many fascinating collective quantum phenomena like superconductivity, magnetism and Bose-Einstein condensation. Although all properties of matter are described by and can in principle be determined from the laws of quantum mechanics. Physicists have very often serious difficulties to understand them in detail and to predict them by starting from fundamental rules and first principles. One reason for these difficulties is the fact that the number of parameters needed to describe a many-particle quantum system grows exponentially with the number of particles. Hence, comparing a theoretical model for the behaviour of more than, say, thirty particles with experimental reality is not possible by simulating the theoretical model numerically on a classical computer without making serious simplifications.

When thinking about this problem of simulating quantum systems on classical computers Richard Feynman came in the early eighties to the conclusion that such a classical simulation typically suffers from an exponential slowdown, whereas another quantum system could in principle do the simulation efficiently with bearable overhead.

In this way a quantum computer operated as a quantum simulator could be used as a link between theoretical models which are formulated on a fundamental level and experimental observations. Similar to Shor's algorithm a quantum simulator such as Deutsh algorithm, Deutch Joza algorithm, Simon's algorithm, and Grover's algorithm, would yield a quadratic to exponential speedup compared to a classical computer (we will discuss these quantum algorithms in the chapter four of this thesis "Quantum computers"). An important difference between these two applications is, however, that a useful Shor-algorithm quantum computer requires thousands of qubits whereas a few tens of qubits could already be useful for the simulation of quantum systems.

However, the crucial question remain is how can a quantum computer be built? On the one hand, progress has been made in recent years in the experimental controlled manipulation of very small quantum systems that can be not called other than spectacular, in a way that was not imaginable not long ago. Quantum gates have been implemented in the base spin manipulation, and with nuclear magnetic resonance techniques, even small quantum algorithms have been realized.Moreover, completely new ways of controlling individual quantum systems will have to be devised, potentially combining different ideas from quantum optics and solid state physics. Any such implementation will eventually have live up to some requirements that have maybe most distinctly been formulated by DiVincenzo as generic requirements in practical quantum computation (we will discuss the DiVincenzo criteria and the implementation of quantum computers in the third chapter “The use of spin in quantum computers”).

Besides the quantum computer with its mentioned applications quantum information science yields a couple of other useful applications which might be easier to realize. The best example is quantum cryptography which enables one to transmit information with “the security of nature’s laws”. However, small building blocks of a quantum computer, i.e., small quantum circuits may be useful as well. One potential application is for instance in precision measurements like in atomic clocks. We also will discuss the link between spin transport and the use of the spin in the build of quantum computers.

I-1- Introduction:

In the last three decades the world of computers and especially that of microprocessors has witnessed an exponential growth in both productivity and performance. The integrated circuit industry has followed a steady path of constantly shrinking devices geometries and increased functionality that larger chips provide. The technology that enabled this exponential growth is a combination of advancements in process technology, microarchitecture, architecture and design and development tools. Together, these performances and functionality improvements have resulted in a history of new technology generations every two to three years, commonly referred to as 'Moore Law'. Each new generation has approximately doubled logic circuit density and increased performance by about 40%. This chapter analyses some of the microarchitectural techniques that are typical for contemporary high-performance microprocessors.

I-2- Evolution of semiconductor technology:

During the past 40 years the semiconductor industry has distinguished it self both by rapid space of performance improvements in its products, and by a steady path of constantly shrinking device geometries and increasing chip size.

Technology scaling has been the primary driver behind improving the performance characteristics of integrated circuits's (IC). The speed and integration density of IC's have dramatically improved. Exploitation of a billion transistor capacity of a single microprocessor requires new system paradigms and significant improvements to design productivity. Structural complexity and functional diversity of such computers are the challenges for the design teams. Structural complexity can be increased by having more productive design methods and by putting more resources in design work. Functional diversity of information technology products will increase too. The next generation products will be based on computers, but the full exploitation of silicon capacity will require drastical improvements in design productivity and system architecture.

Together these performances and functionality improvements are generally identified in a history of new technology generations with the growth of the microprocessor, which is frequently described as a 'Moore's Law'. Moore's Law states that each new generation has approximately doubled logic circuit density and increased performance by 40% while quadrupling memory capacity. According to International Technology Roadmap for Semiconductor (IRTS) projections, the number of transistors per chip and the local clock frequencies for high performance microprocessors will continue to grow exponentially in the next 10 years too. The 2003 IRTS predicts that by 2014 microprocessor gate length will have been 35 nm, voltage will drop to 0.4V, and clock frequency will rise to almost 30 GHz. ***Figure I.1*** presents some of these predictions. As a consequence, experts expect that in the next 10 years the transistor count for microprocessors will increase to 1 billion.

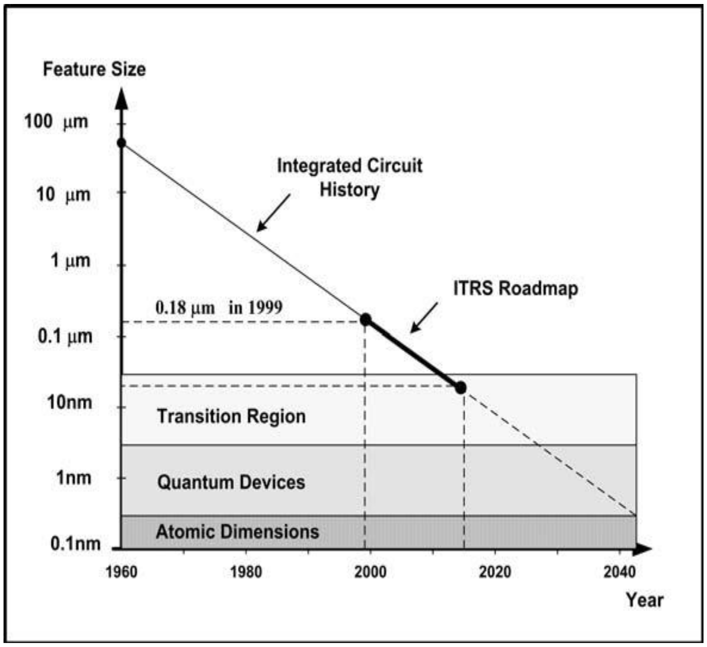


Figure I-1: Trends in future size over time [1]

I-3- Moore's Law

The pace of IC technology over the past forty years has been well characterized by Moore's Law [1]. It was noted in 1965 by Gordon Moore, research director of Fairchild Semiconductor, that the integration density of the first commercial integrated circuit was doubled approximately every year. From the chronology in ***Table I-1***, we see that the first microchip was invented in 1959. Thus, the complexity was one transistor. In 1964, complexity grew up to 32 transistors, and in 1965, a chip in the Fairchild R&D lab had 64 transistors. Moore predicted that chip complexity would be doubled every year based on data for 1959, 1964, and 1965.

year	**Microchip complexly transistors**	**Moore’s Law Complexity: transistors**
1959	1	2^0=1
1964	32	2^5=32
1965	64	2^6=64
1975	64.000	2^{16}=64.000

Table I-1: Complexity of microchip and Moore's law [2]

In 1975, the prediction was revised to suggest a new, slower rate of growth. Doubling of the IC transistors count every two years. This trend of exponential growth of IC complexity is commonly referred to as *Moore's Law I*. However some people say that Moore's Law complexity predicts a doubling every 18 months.

As a result, since the beginning of commercial production of IC's in the early 1960's, circuit complexity has risen from a few transistors to hundreds of billion transistors functioning together on a single monolithic substrate. Furthermore, Moore's law is expected to continue at a comparable pace for at least another decade.

Memory size has also increased rapidly since 1965, when the PDP-8 came with 4 KB of core memory and when an 8 KB system was considered large. In 1981, the IBM PC machine was limited to 640 KB memory. By the early 1990's, 4 or 8 MB memories for PCs were rule, and in 2000, the standard PC memory size grew to 64-128 MB, in 2003 it was in the range from 256 up to 512 MB.

Disk memory has also increased rapidly: from small 32 - 128 kB disks for PDP 8e computer in 1970 to 10 MB disk for the IBM XT PC in 1982. From 1991 to 1997, disk storage capacity increased by about 60% per year, yielding an eighteen folds increase in capacity. In 2001, the standard desktop PC came with a 40 GB hard drive, and in 2003 with 120 GB. If Moore's law predicts a doubling of microprocessor complexity every two years, disk storage capacity will increase by 2.56 times each two years, faster than Moore's Law.

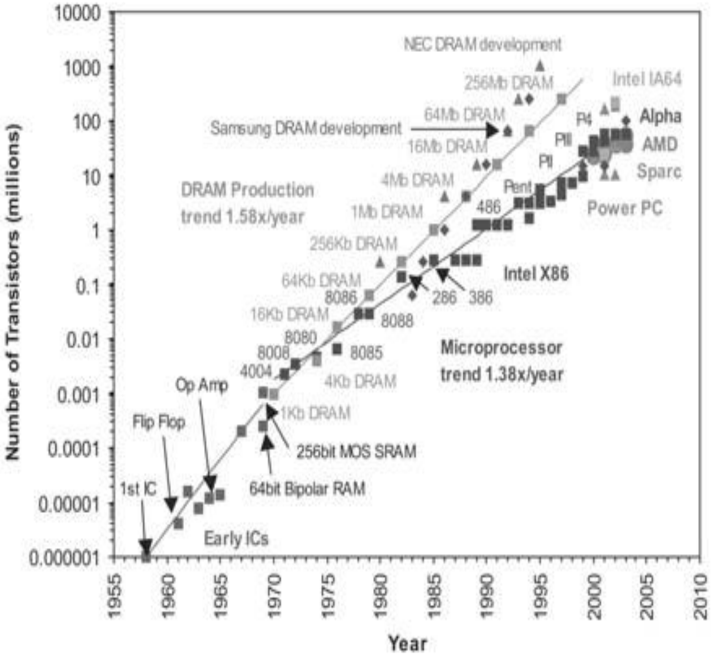


Figure I-2: Evolution of transistor count of CPU/microprocessor and memory ICs [1]

Tendencies in capacity and speed increasing for random logic, DRAM, and disk, during the past period, are given in ***Table I-2***.

	Capacity	Speed (latency)
Logic	2 × in 3 years	2 × in 3 years
DRAM	4 × in 3 years	2 × in 10 years
Disk	4 × in 3 years	2 × in 10 years

Table I-2: Capacity and speed increasing during the past period [2]

I-4- Limits of technology scaling

Improved microprocessor performance results largely from technology scaling, which lets designers increase the level of integration at higher clock frequencies. While current implementations use feature sizes of about 0.25 micron, devices with feature sizes smaller than 0.1 micron are expected in the next few years. Meanwhile, device propagation delay (under constant field assumptions) improves linearly with the decrease in feature size.

Nevertheless, designers face several major technical challenges in the deep-submicron era. The most important is that interconnect delay (especially global interconnect delay) does not scale with feature size. If all three dimensions of a wire are scaled down by the same scaling factor, the interconnect delay remains roughly unchanged. Thus, the interconnect delay decreases far less rapidly than the gate delay and proves more significant in the deep-submicron region.

In an effort to minimize interconnect resistance, modern designs scale interconnects height at a much slower pace than interconnect width. Consequently, the aspect ratio ***(T/Win Figure I-3)*** should rise gradually from 1.8 at present to 3.0 by the year 2012. This shift reduces wire resistance but also increases the effects of line coupling, from 20% at 0.7 micron to 80% at 0.18 micron. Cross talk between adjacent wires will pose a more serious problem, and wire congestion will ultimately determine interconnect delay and power dissipation. Implementations that use more devices in critical paths yet offer less wire congestion and shorter interconnect delays may be preferable to older implementations that simply focus on reducing the number of gate delays.

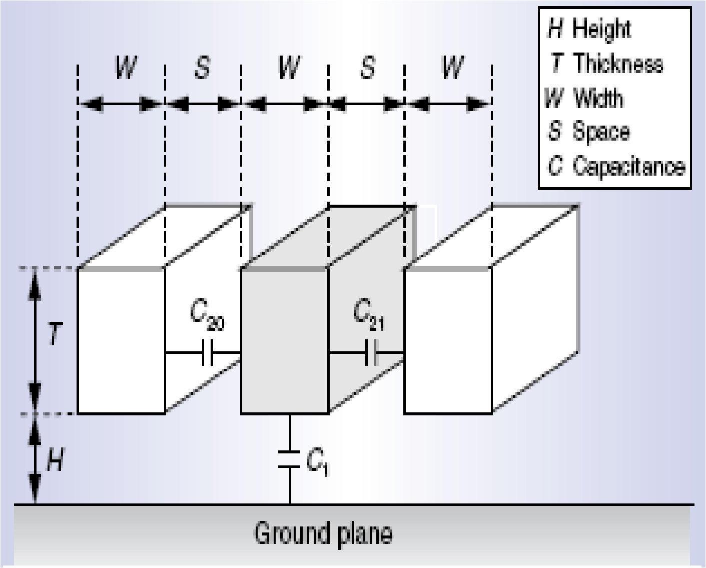


Figure I-3: Interconnect capacitances [6]

Feature sizes below 0.1 micron lead to other technical challenges. High-performance processors need special cooling techniques, which could consume as much as 175 W of power. Enabling GHz signals to travel into and out of the chips requires new circuit designs and algorithms. Preventing latch-up and reducing noise coupling may require new materials such as silicon-on-insulator. Similarly, reducing cross talk and DRAM leakage may require low *k “dielectric insulator”* as well as high *k* materials, respectively. These challenges demand that designers provide whole system solutions instead of treating logic design, circuit design, and packaging as independent phases of the design process.

I-5- Limits in frequency scaling:

Microprocessor performance has improved by approximately 50% per year for the past 15 years. This can be attributed to higher clock frequencies, deeper pipelines, and improved exploitation of instruction-level parallelism (ILP). In the deep-submicron era, we can expect performance improvement to result largely from reducing cycle time at the expense of greater power consumption.

I-5-1- Cycle time:

Processor clock frequencies have increased by approximately 30% per year for the past 15 years, due partly to faster transistors and partly to fewer logic gates per cycle. Traditionally, digital designs have used edge triggered flip-flops extensively. Such a system’s cycle time T_c is determined by $T_c = P_{\max} + C$, where $P_{\max}$ is the maximum delay required for the combinational logic, and C is the total clock overhead, including setup time, clockto-output delay, and clock skew. For high-end server processors, the SIA predicts that the clock cycle time will decrease from roughly 16 FO_4 (fanout-of-four) inverter delays at present to roughly five FO_4 inverter delays at 0.05-micron feature size. As a result, clock overhead takes a significant fraction of the cycle time, and flip-flop clocking systems appear infeasible. Fortunately, a number of circuit techniques can improve microprocessor cycle times:

• Several new flip-flop structures, such as sense-amplifier-based, hybrid latch, and semidynamic flip-flop, have been proposed to lower clock overhead.

• Asynchronous logic eliminates the need for a global clock. Average latency depends on P_{mean} (average logic delay) instead of P_{max} (maximum logic delay), but completion detection and data initialization incur significant overhead.

• Various forms of dynamic logic have been proposed to minimize the effects of clock skew. These techniques reduce clock overhead at the expense of power and scalability.

• Wave pipelining uses P_{min} (minimum logic delay) as a storage element to improve cycle time.

We use wave pipelining to illustrate some of the new clocking considerations. For memories and other functional blocks that contain regular interconnect structures, wave pipelining is an attractive choice. The technique relies on the delay inherent in combinatorial logic circuits. Suppose a given logic unit has a maximum interlatch delay of P_{max} and a corresponding minimum delay of P_{min} with clock overhead C. Then the fastest achievable cycle time t equals P_{max}-$P_{\text{min}} + C$.

As with conventionally clocked systems, system clock rate T_c is the maximum t_i over i latched stages. Sophisticated tools can ensure balanced path delays and thus improve cycle time. In practice, using special tools lets us set P_{min} to within about 80% to 90% of ($P_{\text{max}} + C$). While this would seem to imply clock speedup of more than five times the maximum clock using traditional clocking schemes, environmental issues such as process variation and temperature gradient across a die restrict realizable clock rate speedup to about three times. ***Figure I-4*** details the register-to-register waveforms of a wave-pipelined vector multiplier. The achieved rate shown at the bottom of the figure is more than three times faster than the traditional rate determined by the latency between the multiplier input and output (shown in the top two segments of ***Figure I-4***).

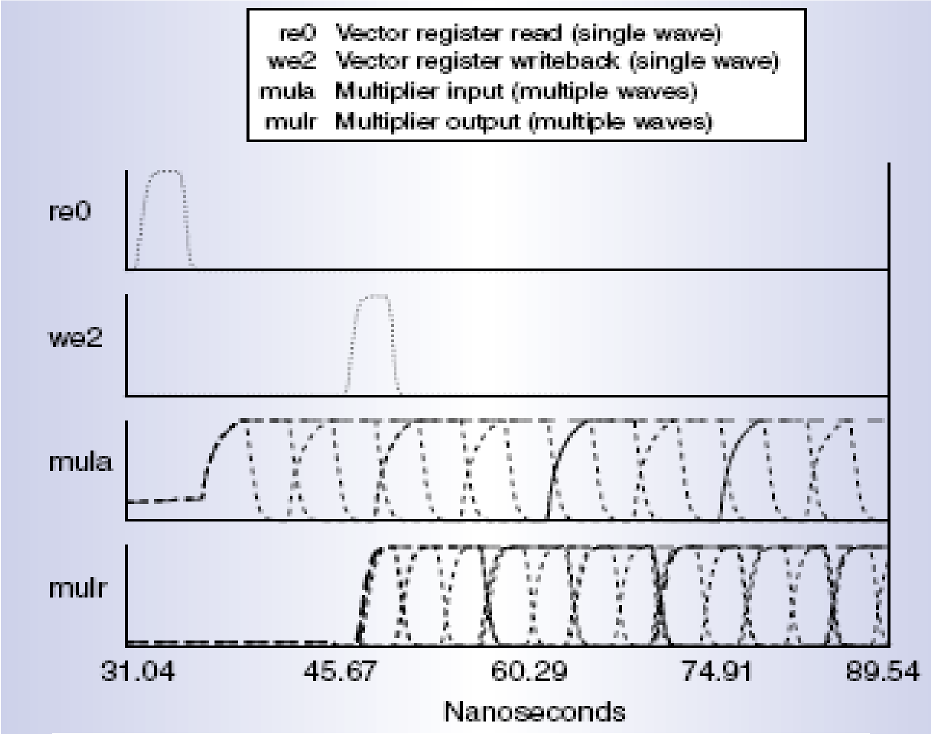


Figure I-4: Wave pipelined vector multiplication [7]

Wave pipelining also exemplifies how aggressive new techniques offer architects and implementers both benefits and challenges. Although allowing significant clock-rate improvements over traditional pipelines, wave pipelines cannot be stalled without losing the in-flight computations. Because individual waves in the pipeline exist only by virtue of circuit delays, they cannot be controlled between pipeline latches. In the case of wave pipelines, architecturally transparent replay buffers can provide the effect of a stall and extend the applicability of wave pipelining to applications that require stalling the pipeline. Other new techniques may not fit in directly with current architectures and may also require special treatment to be generally applicable.

I-6- Limits in low power

Low power consumption is one of the crucial factors determining the success of personal mobile communications and portable computing systems in the fastest growing sectors of the consumer electronics market. Mobile computing system and biomedical implantable devices are just a few examples of electronic devices whose power consumption is a basic constraint to be met, since their operativity in the time domain depends on limited energy storage.

The electronic devices at the heart of such products need to dissipate low power, in order to conserve battery life and meet packaging reliability constraints. Lowering power consumption is important not only for lengthening battery life in portable systems, but also for improving reliability, and reducing heat-removal cost in high performance systems. Consequently, power consumption is a dramatic problem for all integrated circuits designed today.

Low power design in terms of algorithms, architectures, and circuits has received significant attention and research input over the last decade. The implementation can be categorized into system level, algorithm level, architecture level, circuit level, and process/device level.

The system level is the highest layer which strongly influences power consumption and distribution by partitioning system factors.

The algorithm level is the second level, which defines a detailed implementation outline of the required original function, i.e. it determines how to solve the problem and how to reduce the original complexity.

At the architecture level there are still many options and wide freedom in implementation, such as, for example, CPU - microprocessor, DSP (Digital Signal Processor), ASIC (Application Specific Integrated Circuit) - dedicated hardware logic, reconfigurable logic, etc.

The circuit level is the most detailed implementation layer. This level is explained as a module level such as multiplier or memory and basement level like voltage control that affects wide range of the chip.

The process level and the device level are the lowest levels of implementation. This layer itself does not have drastic impact directly. However, when it is oriented towards voltage reduction, this level plays a very important role in power saving.

Present day general purpose microprocessor designers are faced with the daunting task of reducing power dissipation since power dissipation quickly becomes a bottleneck for future technologies. For all the integrated circuits used in battery-powered portable devices, power consumption is the main issue. Furthermore, power consumption is also the main issue for high-performance integrated circuit due to heat dissipation. Consequently, power consumption is a dramatic problem for all integrated circuits designed today.

I-7- Computational integrity

The last basic trade-off is determining the level of computational integrity. When rebooting a personal computer after an application has caused the system to crash, we may wonder about the application or the system or both. However, the observed failure is a retrograde problem solved years ago in hardware with the introduction of user and system states and corresponding memory protection. In looking ahead to improved models of computational integrity, we should consider

- Reliability,
- Testability,
- Serviceability,
- Process recoverability, and
- Fail-safe computation.

Reliability is a characteristic of the implementation media. Circuits and cells may fail, but this need not lead immediately to demonstrable faults in the processor. Indeed, smaller feature sizes may lead to increasing failures over time resulting from electrostatic overstress, and so on. Error correction systems provide an important way to recover from certain modes of device failure. In case of transient errors, error detection systems coupled with instruction retry are a minimum requirement for enabling correct computations.

Testable designs explicitly include accessibility paths, such as scan paths, that enable special validation programs to verify a processor’s correct operation over a broad variety of state combinations. Testability is important for continuing test and design validation.

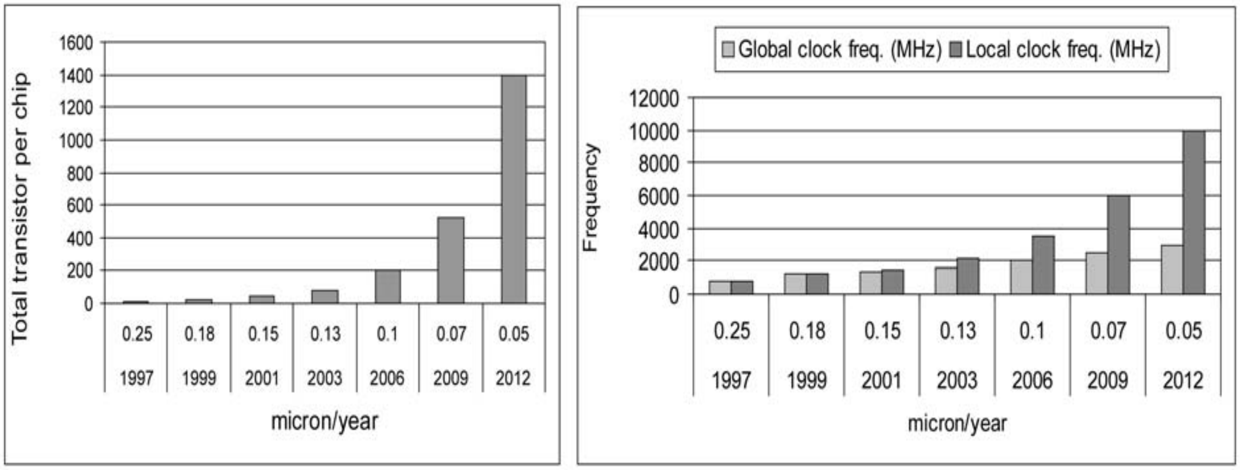
Serviceability allows for ready diagnosis of both transient and permanent failures. It depends on error detection, error scanning on detection, and error logging. The goal is a design that lets us identify degraded paths caused by recoverable but recurring errors.

Process recoverability includes features for instruction retry, process rollback, and, in multiprocessor systems, process migration to another processor.

Fail-safe computation integrates all the above with environmental considerations such as power and temperature. In principle, even power failure should not cause an executing process to abort. Using an uninterruptible power supply or some other backup system lets us save the system state so that computation can resume when power returns.

I-8- Future Directions in Microprocessor Systems

Deep-submicron technology allows billions of transistors on a single die, potentially running at gigahertz frequencies. According to Semiconductor Industry Association projections, the number of transistor per chip and the local clock frequencies for high performance microprocessors will continue to grow exponentially in the near future, as it is illustrated in ***Figure I-5***. This ensures that future microprocessors will become even more complex.



(a) total transistor per chip (b) on - chip local clock

Figure I-5: The National Technology Roadmap For semiconductor: a- total transistor per chip b- On-chip local clock [3]

One approach is to add more memory (either cache or primary) to the chip, but the performances gain from memory alone is limited. Another approach is to increase the level of system integration, bringing support functions like graphics accelerators and I/O controllers on chip. Although integration lowers system costs and communication latency, the overall performance gain to application is again marginal.

In the sequel we will point to some of the new directions oriented towards system/microprocessor performance improvement mainly intended to enhance system/ processor's computational capabilities.

I-8-1- Microprocessor today - microprocessor tomorrow

Microprocessors have gone through significant changes during the last three decades. However, the basic computational model has not been changed much. A program consists of instructions and data. The instructions are encoded in a specific instruction set architecture (ISA). The computational model is still a single instruction stream based on, sequential execution model, operating on the architecture states (memory and registers). It is a job of the microarchitecture, the logic, and the circuits to carry out this instruction stream in the best way.

Figure I-6-a shows the level of transformation that a problem, initially described in some natural languages like English, French or Arabic has to pass through in order to be solved. When we say microprocessor today we generally assume the shaded region of ***Figure I-6-a***, where each microprocessor consists of circuit that implement hardware structure (collectively called the microarchitecture) that provide an interface to the software. As it can be seen from ***Figure I-6-a*** the compiled program uses to tell the microprocessor what it (the program) needs to be done, and the microprocessors use to know what it must be carried out in behalf of the program. The ISA is implemented by a set of hardware structures collectively referred to as the microprocessor's microarchitecture. If we take our levels of transformation and include the algorithm and language into microprocessor, the microprocessor then becomes the thing that uses device technology to solve the problem ***(Figure I-6-b)***.

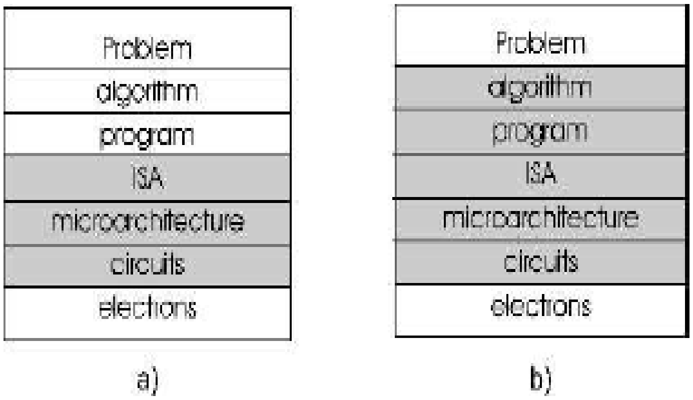


Figure I-6: a- The Microprocessor today ***b-*** The Microprocessor tomorrow [10]

I-8-2- Future directions in microarchitectures

Future microprocessors will be faced with new challenges. Numerous techniques have been proposed. Most of them have multiple sequencers, and are capable of processing multiple instruction streams. In the sequel, we will discuss some microarchitectural techniques that are likely to be used commercially in the near future:

I-8-2-1- Multithreading or multiprocessing:

The processor is composed as a collection of independent processing elements (PEs), each of which executes a separate thread or flow control. By designing the processor as a collection of PEs, (a) the number of global wires is reduced, and (b) very little communication occurs through global wires. Thus, much of communication occurring in the multi- PE processor is local in nature and occurs through short wires. The commonly used model for control flow among threads is the parallel threads model. The *fork* instruction specifies the creation of new threads and their starting addresses, while the *join* instruction serves as a synchronizing point and collects the threads. The thread sequencing model is illustrated in ***Figure I-7***.

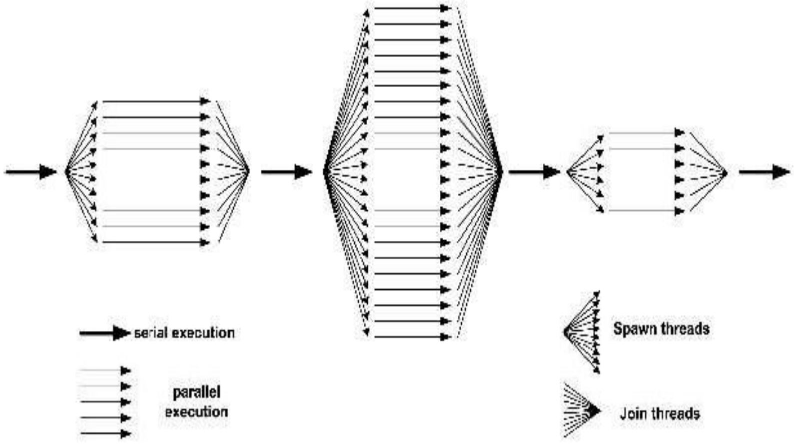


Figure I-7: Parallelism profile for a parallel thread model [11]

I-8-2-2- Simultaneous-multithreading (SMT):

Is a processor design that consumes both thread-level and instruction-level parallelism. In SMT processors thread-level parallelism can come from either multithread, parallel programs or individual, independent programs in a multiprogramming workload. ILP comes from each single program or thread. Because it successfully (and simultaneously) exploits both types of parallelism, SMT processors use resources more efficiently, and both instruction throughput and speedups are greater. ***Figure I-8*** shows how three different architectures partition issue slots (functional units).

The rows of squares represent issue slots. The processor either finds an instruction to execute (filled box) or it allows the slots to remain unused (empty box).

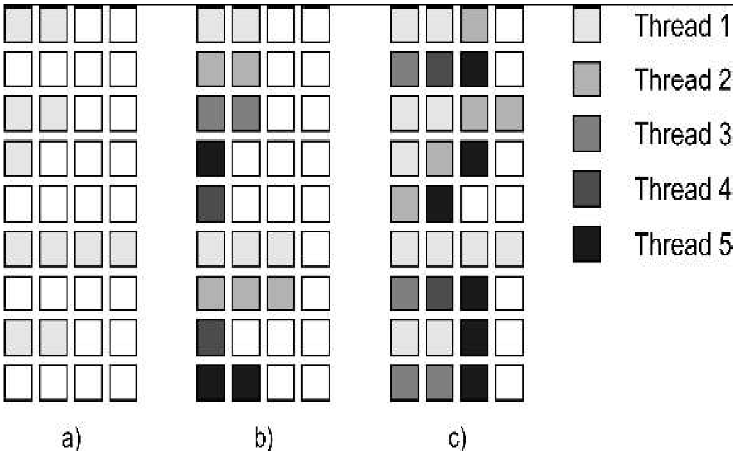


Figure I-8: How three different architectures partition issue slots: a- superscalar b- multithreaded superscalar c- SMT [11]

I-8-2-3- Chip multiprocessor (CMP):

The idea is to put several microprocessors on a single die ***(Figure I-9)***. The performance of small-scale CMP scales close to linear with the number of microprocessors and is likely to exceed the performance of an equivalent multiprocessor system. CMP is an attractive option to use when moving to a new process technology. New process technology allows us to shrink and duplicate our best existing microprocessor on the some silicon die, thus doubling the performance at the same power.

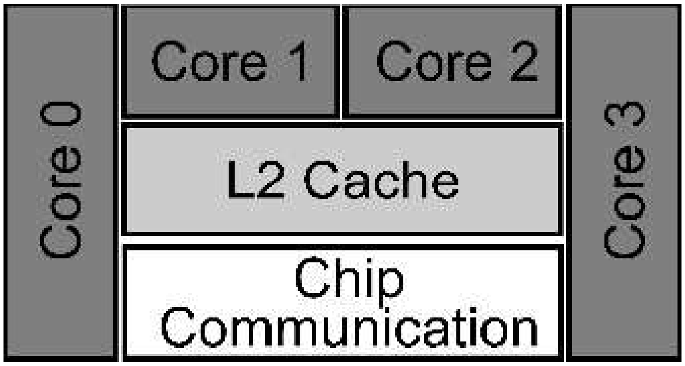


Figure I-9: Chip multiprocessors [11]

I-9- Networks on chips

According to ITRS prediction, by the end of the decade, system on a chip (SoCs) using 50nm transistors and operating below 1V, will grow up to 4 billion transistors running at 10 GHz. The major design problem accompanied with these chips will be the challenge how to provide correct function and reliable operation of the interacting components. On-chip physical interconnections will present a limiting factor for performance, and possibly for energy consumption.

Synchronization of future chips with a single clock source and negligible skew will be extremely difficult, or even impossible. The most likely synchronization paradigm for future chips – globally asynchronous and locally synchronous – involves using many different clocks.

In the absence of a single timing reference, SoC chips become distributed systems on a single silicon substrate. In these solutions, components will initiate data transfer autonomously, according to their needs, i.e. the global communication pattern will be fully distributed.

On-chip networks relate closely to interconnection networks for high performance parallel computers with multiple processors, where processor is an individual chip. Like multiprocessor interconnection networks, nodes are physically closer to each other and have high link reliability. From the design stand point, network reconfigurability will be a key in providing plug-and-play component use because the components will interact with one another through reconfigurable protocols.

I-10- Conclusion:

Human appetite for computation has grown even faster than the processing power that Moore's law predicted. We need even more powerful processors just to keep up with modern applications like interactive multimedia, mobile computing, wireless communications, etc. To make matters more difficult, we need these powerful processors to use less energy than we have been accustomed to, i.e. to design power aware components/systems. To achieve this functionality we must rethink the way we design our contemporary computers. Namely, rather than worrying solely only about performance, we need now to judge computers by their performance, power, cost product. This new way of looking at processors will lead us to new computer architectures and new ways of thinking about computer system design. Thus, if making transistors smaller and smaller is continued with the same rate as in the past years, then by the year of 2020, the width of wire in a computer chip will be no more than a size of a single atom. These are size for which rules of classical physics no longer apply. Computer designed on today's chip technology will not continue to get cheaper and better. Because of its predicted great power, quantum computer is an attractive next step in computer technology. Theoretically, it can run without energy consumption and billion times faster than today's computers. Because quantum computers are based on the principles of quantum theory, let us first review what quantum mechanics is and what quantum information theory is.

II-1- Introduction:

Quantum physics covers a set of physical laws that apply at microscopic scale. While fundamentally different from the majority of laws that appear to apply at our own scale, that laws of quantum physics nevertheless underpin the general basis of physics at all scales. That said, on the macroscopic scale, quantum physics in action appears to behave particularly strangely, except for a certain number of phenomena that were already curious, such as "superconductivity" or "superfluidity", which in fact can only explained by the laws of quantum physics. Quantum physics gets its name from the fundamental characteristics of quantum objects, i.e., characteristics such as the angular momentum (spin) of discrete or discontinuous particles called quanta, which can only take values multiplied by an elementary quantum.

II-2- Basic concepts in quantum mechanics

II-2-1- Wave-particle duality

The Bohr model of the atom involved two puzzling features - the electron was treated as a wave, and light was treated as a particle (a photon). The connection to these new pictures of electrons and light from our more familiar view of an electron as a particle and light as a wave is provided by the relation

$$\lambda = h/_{mv} \quad \dots \qquad \textbf{\textit{(II-1)}}$$

which links the mass (m) and speed (v) of an electron to the wavelength (λ) of the associated wave, and by the relation

$$E = hv \quad \dots \qquad \textbf{\textit{(II-2)}}$$

which links the frequency (ν) of a light wave to the energy (E) of the associated photon.

These relations are not derivable from other relations; they are hypothesized, and are ``true" only so long as they satisfy experimental verification. Even so, they are unlike most mathematical statements, however, in the following sense. A statement like Newton's 2nd law of motion,

$$F = m*v \quad \dots \qquad \textbf{\textit{(II-3)}}$$

refers on both sides of the equal sign to the same object: one says the force on some object is equal to the mass (m) of the object times the acceleration (v) of the object. Implicit in this is the fact that both sides of the equation are referring to the same object, and so we have in mind one common picture of the object. However, the preceding quantum relations are referring to different pictures of an object: on one side of the equation we view the object as a particle, and use words to describe the particle's mass, speed, and energy, and on the other side of the equation we view the object as a wave, and use words like wavelength and frequency to describe that wave.

Questions then arise. What is an electron? Is it a particle or a wave? And what is light? A wave or a photon? The answer to these is found in the statement of wave-particle duality:

All objects exhibit at times a wave like nature, and at other time a particle like nature

Thus, objects (light, electrons, bowling balls, ...) can at times appear to us as waves, and at other times as particles. In this sense they are neither particles nor waves, in an absolute sense, but only exhibit wave or particle properties, depending on the experiment being performed.

II-2-2- Heisenberg's Uncertainty principle:

In quantum physics, the outcome of even an ideal measurement of a system is not deterministic, but instead is characterized by a probability distribution, and the larger the associated standard deviation is, the more "uncertain" we might say that that characteristic is for the system. The Heisenberg uncertainty principle, or Indeterminacy Principle, articulated in 1927 by the German physicist Werner Heisenberg, gives a lower bound on the product of the standard deviations of position and momentum for a system, implying that it is impossible to have a particle that has an arbitrarily well-defined position and momentum simultaneously. More precisely, the products of the standard deviations in each of the three spatial dimensions are bounded by

$$\begin{aligned}(\Delta x)(\Delta p_x) &\geq \hbar/2\\ (\Delta y)(\Delta p_y) &\geq \hbar/2\\ (\Delta z)(\Delta p_z) &\geq \hbar/2\end{aligned} \qquad \ldots \qquad \textbf{\textit{(II-4)}}$$

where $\hbar$ is the reduced Planck constant; Δx, Δy, and Δz are the standard deviations of the three coordinates of position; and Δp_x, Δp_y, and Δp_z are the standard deviations of the three components of momentum. The principle generalizes to many other pairs of quantities besides position and momentum (for example, angular momentum about two different axes), and can be derived directly from the axioms of quantum mechanics.

Note that the uncertainties in question are characteristic of the mathematical quantities themselves. In any real-world measurement, there will be additional uncertainties created by the non-ideal and imperfect measurement process. The uncertainty principle holds true regardless of whether the measurements are ideal (sometimes called von Neumann measurements) or non-ideal (Landau measurements). Note also that the product of the uncertainties, of order 10^{-35} Joule-seconds, is so small that the uncertainty principle has negligible effect on objects of macroscopic scale, despite its importance for atoms and subatomic particles.

As we said, this principle is a consequence of the wave-particle duality. The amplitude of the wave associated with a particle corresponds to its position, and the wavelength (more precisely, its Fourier transform) is inversely proportional to momentum. In order to localize the wave so as to have a sharp peak (i.e., a small position uncertainty), it is necessary to incorporate waves with very short wavelengths, corresponding to high momenta in all directions, and therefore a large momentum uncertainty. A helpful analogy can be drawn between the wave associated with a quantum-mechanical particle and a more familiar wave, the time-varying signal associated with, say, a sound wave. It is meaningless to ask about the frequency spectrum at a single moment in time, because the measure of frequency is the measure of a repetition recurring over a period of time. Indeed, in order for a signal to have a relatively well-defined frequency, it must persist for a long period of time, and conversely, a signal that occurs at a relatively well-defined moment in time (i.e., of short duration) will necessarily encompass a broad frequency band. This is, indeed, a close mathematical analogue of the Heisenberg principle.

II-2-3- Quantum superposition:

The superposition principle plays the most central role in all considerations of quantum information. An important experiment in quantum mechanics is the double slit experiment which determines the quantum superposition principle. The essential ingredients of the experiment are a source, a double-slit assembly, and an observation screen on which we observe interference fringes. These interference fringes may easily be understood on the basis of assuming a wave property of the particles emerging from the source. It might be mentioned here that the double-slit experiment has been performed with many different kinds of particles ranging from photons, via electrons, to neutrons and atoms. Quantum mechanically, tha state is the coherent superposition

$$|\psi\rangle = \frac{1}{\sqrt{2}}\left(|\psi_a\rangle + |\psi_b\rangle\right), \qquad \ldots \qquad \textbf{\textit{(II-5)}}$$

Where $|\psi_a\rangle$ and $|\psi_b\rangle$ (see II-3 to know the notion of brackets $\langle\ |\ \rangle$) describe the quantum state with only slit a or slit b open.

II-2-4- Measurement theory

Suppose we have a system with N distinguishable states $|0\rangle, |1\rangle, \ldots, |N-1\rangle$, and some apparatus that will reliably distinguish these N states. Without loss of generality, let us say the apparatus will output the (classical) label 'i' together with the observed state $|i\rangle$ when $|i\rangle$ is provided as input. In other words, the measurement apparatus provides a classical description of the measurement outcome (which we simply denote as i where we indexed the possible measurement outcomes using the indices i; the values i do not need to be integers), along with some quantum state. Traditionally, the classical description or label is often described as a needle pointing to some value on a dial. But if we assume only finite resolution we can just as well assume a digital display with sufficiently many digits.

Quantum mechanics tells us that if the state $\sum_i \alpha_i |i\rangle$ is provided as input to this apparatus, it will output label *i* with probability $|\alpha_i|^2$ and leave the system in state $|i\rangle$.

Thus, for a given orthonormal basis $B = \{|\varphi_i\rangle\}$ of a state space H_A for a system *A*, it is possible to perform a *Von Neumann measurement* on system H_A with respect to the basis *B* that, given a state

$$|\psi\rangle = \sum_i \alpha_i |\varphi_i\rangle \qquad \ldots \qquad \textbf{\textit{(II-6)}}$$

outputs a label *i* with probability $|\alpha_i|^2$ and leaves the system in state $|\varphi_i\rangle$.

Furthermore, given a state $|\psi\rangle = \sum_i \alpha_i |\varphi_i\rangle$ from a bipartite state space $H_A \otimes H_B$ (the $|\varphi_i\rangle$ are orthonormal; the $|\gamma_i\rangle$ have unit norm but are not necessarily orthogonal), then performing a Von Neumann measurement on system *A* will yield outcome *i* with probability $|\alpha_i|^2$ and leave the bipartite system in state $|\varphi_i\rangle\,|\gamma_i\rangle$.

For the state $|\psi\rangle = \sum_i \alpha_i |\varphi_i\rangle$, note that $\alpha_i = \langle\varphi_i\|\psi\rangle = \langle\varphi_i|\psi\rangle$, and thus

$$|\alpha_i|^2 = \alpha_i^* \alpha_i = \langle\psi|\varphi_i\rangle\langle\varphi_i|\psi\rangle \qquad \ldots \qquad \textbf{\textit{(II-7)}}$$

We can see that two states $|\psi\rangle$ and $e^{i\theta}|\psi\rangle$ (differing only by a global phase) are equivalent. Consider the state $e^{i\theta}|\psi\rangle = \sum_i \alpha_i e^{i\theta}|\psi_i\rangle$ immediately before a measurement. The result *i* will occur with probability

$$p\,(i) = \alpha^*{}_i\, e{-}^{i\theta}\alpha_i e^{i\theta} = \alpha^*{}_i\, \alpha_i = |\alpha_i|^2 \ldots \qquad \textbf{\textit{(II-8)}}$$

and thus the resulting probability is the same as it would be for the state $|\psi\rangle$.

The statistics of any measurements we could perform on the state $e^{i\theta}|\psi\rangle$ are exactly the same as they would be for the state $|\psi\rangle$. This explains that global phases have no physical significance.

Combining the Measurement Postulate above with the other postulates, we can derive more general notions of measurement. In particular, if one wishes to measure a pure state $|\psi\rangle$one can add an ancillary register of arbitrary size initialized to some fixed state, say$|0,0,\ldots 0\rangle$. One can then perform a unitary operation on the joint system, followed by a Von Neumann measurement on some subsystem of the joint system to obtain a label *i*. Depending on what is done with the rest of the system (i.e. the part of the system that was not measured), one can derive a variety of generalized notions of quantum measurement.

A Von Neumann measurement is a special kind of *projective measurement*.

An orthogonal projection is an operator *P* with the property that $P^{\dagger} = P$ and $P^2 = P$. For any decomposition of the identity operator $I = \sum_i P_i$ into orthogonal projectors P_i, there exists a projective measurement that outputs outcome *i* with probability $p(i)$ = $\langle\psi|P_i|\psi\rangle$and leaves the system in the renormalized state $\frac{P_i|\psi\rangle}{\sqrt{p(i)}}$. In other words, this measurement projects the input state$|\psi\rangle$ into one of the orthogonal subspaces corresponding to the projection operators P_i, with probability equal to the square of the size of the amplitude of the component of $|\psi\rangle$ in that subspace.

Note that the Von Neumann measurement as described in the Measurement Postulate (which can be described as a ‘complete’ or ‘maximal’ measurement) is the special case of a projective measurement where all the projectors P_i have rank one (in other words, are of the form $|\psi_i\rangle\langle\psi_i|$for a normalized state$|\psi_i\rangle$).

Projective measurements are often described in terms of an *observable*. An observable is a Hermitean operator *M* acting on the state space of the system. Since *M* is Hermitean, it has a spectral decomposition

$$M = \sum_i m_i P_i \quad \ldots \qquad \textbf{\textit{(II-9)}}$$

where P_i is the orthogonal projector on the eigenspace of *M* with real eigenvalue m_i. Measuring the observable corresponds to performing a projective measurement with respect to the decomposition $I = \sum_i P_i$ where the measurement outcome *i* corresponds to the eigenvalue m_i.

II-2-5- Entanglement

Consider a source which emits a pair of particles such that one particle emerges to the left and the other one to the right (see source S in ***Figure II-1***). The source is such that the particles are emitted with opposite momenta. If the particle emerging to the left, which we call particle 1, is found in the upper beam, then particle 2 traveling to the right is always found in the lower beam. Conversely, if particle 1 is found in the lower beam, then particle 2 is always found in the upper beam. In our qubit language we would say that the two particles carry different bit values. Either particle 1 carries “0” and then particle 2 definitely carries “1”, or vice versa. Quantum mechanically this is a two-particle superposition state of the form

$$\frac{1}{\sqrt{2}}\left(|0\rangle_1 |1\rangle_2 + e^{i\chi} |1\rangle_1 |0\rangle_2\right) \qquad \ldots \qquad \textbf{\textit{(II-10)}}$$

The phase χ is just determined by the internal properties of the source and we assume for simplicity $\chi = 0$. Equation ***(II-10)*** described what is called an entangled state. The interesting property is that neither of the two qubits carries a definite value, but what is known from the quantum state is that as soon as one of the two qubits is subject to a measurement, the result of this measurement being completely random, the other one will immediately be found to carry the opposite value. In a nutshell this is the conundrum of quantum non-locality, since the two qubits could be separated by arbitrary distances at the time of the measurement.

A most interesting situation arises when both qubits are subject to a phase shift and to Hadamard transformation (see II-4-3) as shown in ***Figure II-1*** [24]. Then, for detection events after both Hadamard transformations, that is, for the case of the two particle interferometer verification for detection behind the beamspliters, interesting non local correlations result which violate Bell’s inequalities [24].

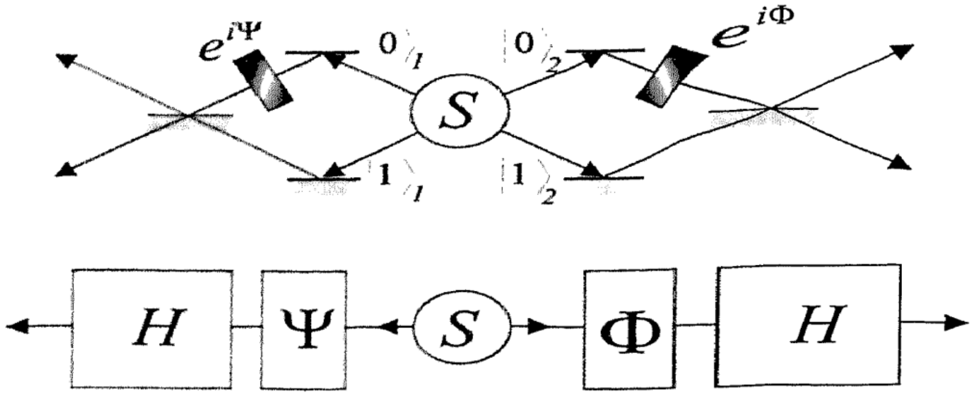


Figure II-1: A source of emits two qubits in an entagled state. Top: A two particle interferometer verification. Bottom: The principle in terms of one photon gates. [24]

The essence of such a violation is that there is no possibility to explain the correlations between the two sides on the basis of local properties of the qubits alone. The quantum correlations between the two sides cannot be understood by assuming that the specific detector on one given side which registers the particle is not influenced by the parameter setting, that is, by the choice of the phase for the other particle.

A very interesting, and for quantum computation quite relevant generalization follows if entanglement is studied for more than two qubits. Consider the simple case of entanglement between three qubits, as shown in ***Figure II-2***. We assume that a source emits three particles, one into each of the apparatuses shown, in the specific superposition, a so-called Greenberger-Horne-Zeilinger (GHZ) [24],

$$\frac{1}{\sqrt{2}}\left(|0\rangle_1|0\rangle_2|0\rangle_3+|1\rangle_1|1\rangle_2|1\rangle_3\right) \qquad \dots \qquad \textbf{\textit{(II-11)}}$$

This quantum state has some very peculiar properties. Again, as in two-particle entanglement, none of the three qubits carries any information on its own; none of them has a defined bit value. But, as soon as one of the three is measured, the other two will assume a well-defined value as long as the measurement is performed in the chosen 0-1 basis. This conclusion holds independent of the special separation between the three measurements.

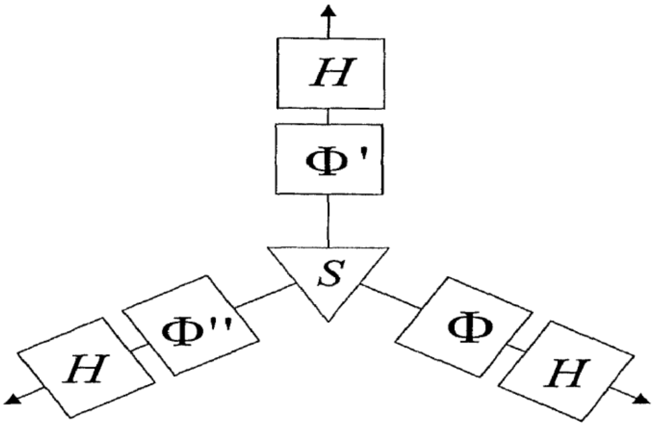


Figure II-2 : Three particles entanglement in a GHZ state. [24]

Most interestingly, if one looks at the relations predicted by the GHZ state ***(II-11)*** between the three measurements after passing the phase shifters and the Hadamard transforms, a number of perfect correlations still result for certain joint settings of the three parameters, the interesting property now being that it is not possible to understand even the perfect correlations with a local model. This shows that quantum mechanics is at variance with a classical local world view not only for the sector of statistical predictions of the theory but also for predictions which can be made with certainty.

II-2-6- Entanglement and quantum indistinguishability

In order to understand both the nature of entanglement and ways of producing it, one has to realize that in states of the general form the equations ***(II-10)*** and ***(II-11)***, we have a superposition between product states. We recall from the discussion of the double-slit diffraction phenomenon that superposition means that there is no way to tell which of the two possibilities forming the superposition actually pertains. This rule must also be applied to the understanding of quantum entanglement. For example, in the state

$$|\Psi\rangle_{12} = \frac{1}{\sqrt{2}}\left(|0\rangle_1 |1\rangle_2 + |1\rangle_1 |0\rangle_2\right) \qquad \cdots \qquad \textbf{\textit{(II-12)}}$$

there is no way of telling whether qubit 1 carries the value”0” or “1”, and likewise whether qubit 2 carries the value ”0” or “1”. Yet if one qubit is measured the other one immediately assumes a well-defined quantum state.

These observations lead us directly to the conditions of how to produce and observe entangled quantum states.

To produce entangled quantum states, one has various possibilities. Firstly, one can create a source which, through its physical construction, is such that the quantum states emerging already have the indistinguishability feature discussed before. This is realized, for example, by the decay of a spin-0 particle into two spin-1/2 particles under conservation of the internal angular momentum . In this case, the two spins of the emerging particles have to be opposite, and, if no further mechanisms exist which permit us to distinguish the possibilities right at the source, the emerging quantum state is

$$|\Psi\rangle_{12} = \frac{1}{\sqrt{2}}\left(|\uparrow\rangle_1|\downarrow\rangle_2 - |\downarrow\rangle_1|\uparrow\rangle_2\right) \quad \dots \qquad \textbf{\textit{(II-13)}}$$

Where, e.g. $|\uparrow\rangle_1$ means particle 1 with spin up. The state ***(II-13)*** has the remarkable property that it is rotationally invariant, i.e., the two spins are anti-parallel along whichever direction we choose to measure.

A second possibility is tat a source might actually produce quantum states of the form of the individual components in the superposition of ***(II-13)***, but the states might still be distinguishable in some way. This happens, for example, in type-II parametric down, where along a certain chosen direction the two emerging photon states are.

$$|H\rangle_1|V\rangle_2 \text{ and } |V\rangle_1|H\rangle_2 \quad \dots \qquad \textbf{\textit{(II-14)}}$$

That means that either photon 1 is horizontally polarized and photon 2 is vertically polarized, or photon 1 is vertically polarized and photon 2 is horizontally polarized. Yet because of the different speeds of light for the H and V polarized photons inside the down-conversion crystal, the time correlation between the two photons is different in the two cases. Therefore, the two terms in ***(II-14)*** can be distinguished by a time measurement and no entangled state results because of this potential to distinguish the two cases. However, in this case too one can still produce entanglement by shifting the two photon-wave packets after their production relative to each other such that they become indistinguishable on the basis of their positions in time.

What this means is the application of a quantum eraser technique where a marker, in this case the relative time ordering, is erased such that we obtain quantum indistinguishability resulting in the state.

$$|\Psi\rangle_{12}=\frac{1}{\sqrt{2}}\left(|H\rangle_1|V\rangle_2+e^{i\chi}|V\rangle_1|H\rangle_2\right) \quad \ldots \qquad \textbf{\textit{(II-15)}}$$

which is entangled.

A third means of producing entangled states is to project a non-entangled state onto an entangled one. We remark, for example, that an entangled state is never orthogonal to any of its components. Specifically, consider a source producing the non-entangled state

$$|0\rangle_1|1\rangle_2 \quad \ldots \qquad \textbf{\textit{(II-16)}}$$

Suppose this state is now sent through a filter described by the projection operator

$$P=|\Psi\rangle_{12}\langle\Psi|_{12,} \qquad \ldots \qquad \textbf{\textit{(II-17)}}$$

Where $|\Psi\rangle_{12}$ is the state of ***(II-12)***. Then the following entangled state results:

$$\frac{1}{2}\left(|0\rangle_1|1\rangle_2+|1\rangle_1|0\rangle_2\right)\left(\langle 0|_1\langle 1|_2+\langle 1|_1\langle 0|_2\right)|0\rangle_1|1\rangle_2=\frac{1}{2}\left(|0\rangle_1|1\rangle_2+|1\rangle_1|0\rangle_2\right), \ldots \textbf{\textit{(II-18)}}$$

It is no longer normalized to unity because the projection procedure implies a loss of qubits.

While each of three methods discussed above can in principle be used to produce outgoing entangled states, a further possibility exists to produce entanglement upon observation of a state. In general, this means that we have an unentangled or partially entangled state of some form and the measurement procedure itself is such that it projects onto an entangled state, in much the same way as discussed just above. This procedure was used, for example, in the first experimental demonstration of GHZ entanglement of three photons .

II-2-7- Decoherence:

Decoherence is a phenomenon that plays a great role in many of the events of quantum mechanics. Understanding decoherence is essential to understanding how classical physics emerges from quantum mechanics.

The basic idea is this: a quantum system, A, in isolation, behaves in a characteristically quantum-mechanical fashion, exhibiting interference effects that reflect the phase difference between the various components of its state vector. For example, if A consists of an electron in a state that is a superposition of equal parts spin up and spin down, there will be measurements that can be performed on the electron that will be sensitive to the phase relationship between these two components. This is quite different from the classical notion of probability: there isn't merely a 50% chance for the electron's spin to be up or down; rather, both possibilities exist simultaneously, and the phase describes a relationship between them that would be meaningless if either was absent.

If system A then interacts with another system, B, in such a manner that different components of A's state vector influence B differently, the two systems become entangled, and observations on A alone will no longer exhibit quantum effects. System A appears to have “collapsed” down to just one component of its original state vector. In the example of the electron, it now acts as if there were merely a 50/50 chance for its spin to be either purely up or purely down.

However, no such “collapse” has really taken place. Measurements on the combined system, A+B, reveal that it is in a pure quantum state, and none of the original components of A's state vector have been lost. Classical physics emerges, essentially, from the inability to observe everything we'd need to in order to detect quantum phenomena in the world at large.

II-3- Linear Algebra and the Dirac notation

II-3-1- The Dirac Notation and Hilbert Spaces

In the Dirac notation, the state “a”of a system is denoted by the ‘ket’ $|a\rangle$. We denote the dual vector for a with a ‘bra’, written as $\langle a|$. Then inner products will be written as ‘bra-kets’ (e.g.$\langle a|b\rangle$).

The kets belong to a vector spaces called *Hilbert* spaces. We will use *H* to denote such a space.

Since *H* is finite-dimensional, we can choose a basis and alternatively represent vectors (kets) in this basis as finite column vectors, and represent operators with finite matrices. The Hilbert spaces of interest for quantum computing will typically have dimension 2^n for some positive integer *n*. This is because, as with classical information, we will construct larger state spaces by concatenating a string of smaller systems, usually of size two.

We will often choose to fix a convenient basis and refer to it as the *computational basis*. In this basis, we will label the 2^n basis vectors in the Dirac notation using the binary strings of length *n*:

$$\underbrace{|00..00\rangle}_{n}, |00...01\rangle, \ldots, |11...10\rangle, |11...11\rangle. \quad \ldots \qquad \textbf{\textit{(II-19)}}$$

The standard way to associate column vectors corresponding to these basis vectors is as follows:

$$|00..00\rangle \Leftrightarrow \begin{pmatrix} 1 \\ 0 \\ 0 \\ . \\ . \\ 0 \\ 0 \end{pmatrix}, |00...01\rangle \Leftrightarrow \begin{pmatrix} 0 \\ 1 \\ 0 \\ . \\ . \\ 0 \\ 0 \end{pmatrix}, \ldots, |11...10\rangle \Leftrightarrow \begin{pmatrix} 0 \\ 0 \\ 0 \\ . \\ . \\ 1 \\ 0 \end{pmatrix}, |11...11\rangle \Leftrightarrow \begin{pmatrix} 0 \\ 0 \\ 0 \\ . \\ . \\ 0 \\ 1 \end{pmatrix} \quad \ldots \textbf{\textit{(II-20)}}$$

An arbitrary vector in *H* can be written either as a weighted sum of the basis vectors in the Dirac notation, or as a single column matrix.

II-3-2- Dual Vectors

For vectors over the complex numbers, an inner product is a function which takes two vectors from the same space and evaluates to a single complex number. We write the inner product of vector ***v*** with ***w*** as $\langle v|w\rangle$. An inner product is such a function having the following properties.

1. Linearity in the second argument

$$\left\langle v, \sum_i \lambda_i w_i \right\rangle = \sum_i \lambda_i \langle v, w_i \rangle \ \ldots \qquad \textbf{\textit{(II.21)}}$$

2. Conjugate-commutativity

$$\langle v, w \rangle = \langle w, v \rangle^* \qquad \ldots \qquad \textbf{\textit{(II.22)}}$$

3. Non-negativity

$$\langle v, v \rangle \geq 0 \qquad \ldots \qquad \textbf{\textit{(II.23)}}$$

with equality if and only if ***v*** = **0**.

A familiar example of an inner product is the *dot product* for column vectors. The dot product of **v** with **w** is written **v** · **w** and is defined as follows.

$$\begin{pmatrix} v_1 \\ v_2 \\ \cdot \\ \cdot \\ v_n \end{pmatrix} . \begin{pmatrix} w_1 \\ w_2 \\ \cdot \\ \cdot \\ w_n \end{pmatrix} = \left(v_1^* v_2^* \ldots v_n^* \right) \begin{pmatrix} w_1 \\ w_2 \\ \cdot \\ \cdot \\ w_n \end{pmatrix} = \sum_{i=1}^{n} v_i^* w_i \qquad \ldots \qquad \textbf{\textit{(II.24)}}$$

Definition II-1:

Let H be a Hilbert space. The Hilbert space H is defined as the set of linear maps H*→C.

We denote elements of H^* by $\langle \chi |$, where the action of $\langle \chi |$ is:

$$\langle \chi | : | \psi \rangle \mapsto \langle \chi | \psi \rangle \in C \qquad \ldots \qquad \textbf{\textit{(II-25)}}$$

Where $\langle \chi | \psi \rangle$ is the inner-product of the vector $| \chi \rangle \in H$ with the vector $| \psi \rangle \in H$. The set of maps H^* is a complex vector space itself, and is called the *dual* vector space associated with *H*. The vector $\langle \chi |$ is called the *dual* of $| \chi \rangle$. In terms of the matrix representation, $\langle \chi |$ is obtained from $| \chi \rangle$ by taking the corresponding row matrix, and then taking the complex conjugate of every element (i.e. the 'Hermitean conjugate' of the column matrix for $| \chi \rangle$). Then the inner product of $| \psi \rangle$ with $| \varphi \rangle$ is $\langle \psi | \varphi \rangle$, which in the matrix representation is computed as the single element of the matrix product of the row matrix representing $| \psi \rangle$ with the column matrix representing $| \varphi \rangle$. This is equivalent to taking the dot product of the column vector associated with $|\psi\rangle$ with the column vector associated with $|\varphi\rangle$.

Two vectors are said to be *orthogonal* if their inner product is zero. The *norm* of a vector $|\psi\rangle$, denoted $\||\psi\rangle\|$ is the square root of the inner product of $|\psi\rangle$ with itself. That is,

$$\||\psi\rangle\| = \sqrt{\langle\psi|\psi\rangle} \qquad \ldots \qquad \textbf{\textit{(II-26)}}$$

The quantity $\||\psi\rangle\|$ is called the *Euclidean norm* of $|\psi\rangle$. A vector is called a *unit vector* if it has norm 1. A set of unit vectors that are mutually orthogonal is called an *orthonormal* set.

Definition II-2:

Consider a Hilbert space H of dimension 2^n. *A set of* 2^n *vectors B =* $\{|b_m\rangle\} \subseteq H$ *is called an* orthonormal basis *for H if*

$$\langle b_n|b_m\rangle = \delta_{n,m} \qquad \forall b_m, b_n \in B \quad \ldots \qquad \textbf{\textit{(II-27)}}$$

and every $|\psi\rangle \in$ *H can be written as*

$$|\psi\rangle = \sum_{b_n \in B} \psi_n |b_n\rangle \qquad \text{for some } \psi_n \in C \quad \ldots \qquad \textbf{\textit{(II-28)}}$$

The values of ψ_n *satisfy* $\psi_n = \langle b_n|\psi\rangle$, *and are called the 'coefficients of* $|\psi\rangle$ *with respect to basis* $\{|b_n\rangle\}$.

Note that if we express $|\psi\rangle = \sum_i \alpha_i |\phi_i\rangle$ with respect to any orthonormal basis $\{|\phi_i\rangle\}$, then $\||\psi\rangle\| = \sum_i |\alpha_i|^2$.

Theorem II-1 [23]

The set $\{\langle b_n|\}$ *is an orthonormal basis for* H^* *called the dual basis.*

II-3-3- Operators

Definition II-3

A linear operator on a vector space H is a linear transformation T : H → H of the vector space to itself (i.e. it is a linear transformation which maps vectors in H to vectors in H).

Just as the inner product of two vectors $|\psi\rangle$ and $|\varphi\rangle$ is obtained by multiplying $|\psi\rangle$ on the left by the dual vector $\langle\varphi|$, an *outer product* is obtained by multiplying $|\psi\rangle$ *on the right* by $\langle\varphi|$. The meaning of such an outer product $|\psi\rangle\,\langle\varphi|$ is that it is an operator which, when applied to $|\gamma\rangle$, acts as follows.

$$\big(|\psi\rangle\langle\varphi|\big)\ |\gamma\rangle = |\psi\rangle\big(\langle\varphi|\gamma\rangle\big) = \big(\langle\varphi|\gamma\rangle\big)\ |\psi\rangle. \quad \ldots \qquad \textbf{\textit{(II-29)}}$$

The outer product of a vector $|\psi\rangle$ with itself is written $|\psi\rangle\langle\psi|$ and defines a linear operator that maps

$$|\psi\rangle\langle\psi\|\varphi\rangle \mapsto |\psi\rangle\langle\psi|\varphi\rangle = \langle\psi|\varphi\rangle|\psi\rangle \qquad \ldots \qquad \textbf{\textit{(II-30)}}$$

That is, the operator $|\psi\rangle\langle\psi|$ projects a vector $|\varphi\rangle$ in *H* to the 1-dimensional subspace of *H* spanned by $|\psi\rangle$. Such an operator is called an *orthogonal projector.*

Theorem II-2 [23]

Let B = $\{|b_n\rangle\}$ *be an orthonormal basis for a vector space H. Then every linear operator T on H can be written as*

$$T = \sum_{b_n, b_m \in B} T_{n,m} |b_n\rangle\langle b_m| \qquad \ldots \qquad \textbf{\textit{(II-31)}}$$

Where $T_{n,m} = \langle b_n|T|b_m\rangle$.

We know that the set of all linear operators on a vector space *H* forms a new complex vector space *L*(*H*) ('vectors' in *L*(*H*) are the linear operators on *H*).

Notice that ***Theorem II-2*** essentially constructs a basis for *L*(*H*) out of the given basis for *H*. The basis vectors for *L*(*H*) are all the possible outer products of pairs of basis vectors from *B*, that is $\{|b_n\rangle\langle b_m|\}$.

The action of *T* is then

$$T : |\psi\rangle \mapsto \sum_{b_n, b_m \in B} T_{n,m} |b_n\rangle\langle b_m|\psi\rangle = \sum_{b_n, b_m \in B} T_{n,m} \langle b_m|\psi\rangle|b_n\rangle. \quad \ldots \qquad \textbf{\textit{(II-32)}}$$

In terms of the matrix representation of *T*, $T_{n,m}$ is the matrix entry in the n^{th} row and m^{th} column.

For any orthonormal basis *B* = $\{|b_n\rangle\}$, the identity operator can be written as

$$1 = \sum_{b_n \in B} |b_n\rangle\langle b_n| \quad \ldots \qquad \textbf{(II-33)}$$

Equation ***(II-33)*** is called the *resolution of the identity in the basis B.*

Notice that, for an operator *T* on *H*, and $|\psi\rangle \in H$, the map

$$|\psi\rangle \mapsto \langle\varphi|(T|\psi\rangle) \quad \ldots \qquad \textbf{(II-34)}$$

is a linear map from *H* to C, and thus belongs to H^*. Each map in H^* corresponds to some vector $\langle\varphi^*|$. The *adjoint* of the operator *T*, denoted T^+, is defined as the linear map that sends $|\varphi\rangle \mapsto |\varphi^*\rangle$, where $\langle\phi|(T|\psi\rangle) = \langle\phi^*|\psi\rangle$ for all $|\psi\rangle$.

Definition II-4

Suppose T is an operator on H. Then the adjoint *of T, denoted T^+, is defined as that linear operator on H* that satisfies*

$$(\langle\psi|T^+|\varphi\rangle)^* = \langle\varphi|T|\psi\rangle \quad , \quad \forall|\psi\rangle, |\varphi\rangle \in H \quad \ldots \qquad \textbf{(II-35)}$$

In the standard matrix representation, the matrix for T^+ is the complex conjugate transpose (also called the 'Hermitean conjugate', or 'adjoint') of the matrix for *T*.

Definition II-5

An operator U is called unitary if $U^+ = U^{-1}$, where U^{-1} is the inverse of U. Note that $U^+ = U^{-1}$ implies $U^+U = I$, where I is the identity operator. The unitary operators preserve inner products between vectors, and in particular, preserve the norm of vectors.

We also define a class of operators that describes the Hamiltonian of a system as well as the observables, which correspond to an important type of measurement in quantum mechanics.

Definition II-6

An operator T in a Hilbert space H is called Hermitian (*or* self-adjoint) *if*

$$T^+ = T \quad \ldots \qquad \textbf{(II-36)}$$

(i.e. it is equal to its own Hermitian conjugate).

Definition II-7

A projector on a vector space H is a linear operator P that satisfies $P^2 = P$. An orthogonal projector is a projector that also satisfies $P^+ = P$.

Definition II-8

A vector $|\psi\rangle$ is called an eigenvector *of an operator T if*

$$T|\psi\rangle = c|\psi\rangle \qquad \dots \qquad \textbf{\textit{(II-37)}}$$

for some constant c. The constant c is called the eigenvalue of T corresponding to the eigenvector $|\psi\rangle$.

Theorem II-3 [23]

If $T = T^\dagger$ and if $T|\psi\rangle = \lambda|\psi\rangle$ then $\lambda \in$ R. In other words, the eigenvalues of a Hermitian operator are real.

Definition II-9

The trace *of an operator A acting on a Hilbert space H is*

$$Tr(A) = \sum_{b_n} \langle b_n | A | b_n \rangle \qquad \dots \qquad \textbf{\textit{(II-38)}}$$

where $\{|b_n\rangle\}$ is any orthonormal basis for H.

II-3-4- The spectral theorem

The *spectral theorem* is a central result in linear algebra, because it is often very convenient to be able to specify a basis in which a given operator is diagonal (i.e. to *diagonalize* the operator). The spectral theorem applies to a wide class of operators which we now define.

Definition II-10

A normal *operator A is a linear operator that satifies*

$$AA^\dagger = A^\dagger A \qquad \dots \qquad \textbf{\textit{(II-39)}}$$

Notice that both unitary and Hermitean operators are normal. So, most of the operators that are important for quantum mechanics and quantum computing are normal.

Theorem II-4 [23]

For every normal operator T acting on a finite-dimensional Hilbert space H, there is an orthonormal basis of H consisting of eigenvectors $|T_i\rangle$ *of T.*

Note that *T* is diagonal in its own eigenbasis: $T=\sum_i T_I|T_I\rangle\langle T_I|$, where T_i are the eigenvalues corresponding to the eigenvectors $|T_i\rangle$. We sometimes refer to *T* written in its own eigenbasis as the *spectral decomposition* of *T*. The set of eigenvalues of *T* is called the *spectrum* of *T*.

The Spectral Theorem tells us that we can always diagonalize normal operators (in finite dimensions). In the linear algebra the diagonalization can be accomplished by a change of basis (to the basis consisting of eigenvectors). The change of basis is accomplished by conjugating the operator *T* with a unitary operator *P*. With respect to the matrix representation for the operator *T*, we can restate the Spectral Theorem in a form which may be more familiar.

Theorem II-5 [23]

For every finite-dimensional normal matrix T, there is a unitary matrix P such that $T = P\Lambda P^{+}$, *where* Λ *is a diagonal matrix.*

The diagonal entries of Λ *are the eigenvalues of T, and the columns of P encode the eigenvectors of T.*

II-3-5- Functions of operators

One of the reasons why the Spectral Theorem is important is that it allows us to simplify the expressions for functions of operators. By the Spectral Theorem, we can write every normal operator *T* in the diagonal form

$$T=\sum_i T_i|T_i\rangle\langle T_i| \qquad \ldots \qquad \textbf{\textit{(II-40)}}$$

First, note that since each $|T_i\rangle\langle T_i|$ is a projector,

$$\left(|T_i\rangle\langle T_i|\right)^m=|T_i\rangle\langle T_i| \qquad \ldots \qquad \textbf{\textit{(II-41)}}$$

for any integer *m*. Also noting that the eigenvectors are orthonormal, we have

$$\langle T_i|T_j\rangle=\delta_{i,j} \qquad \ldots \qquad \textbf{\textit{(II-42)}}$$

So this means that computing a power of T (in diagonal form) is equivalent to computing the powers of the diagonal entries of T:

$$\left(\sum_i T_i |T_i\rangle\langle T_i|\right)^m = \sum_i T_i^m |T_i\rangle\langle T_i| \qquad \ldots \qquad \textbf{\textit{(II-43)}}$$

The Taylor series for a function f : C → C, has the form

$$f(x) = \sum_{m=0}^{\infty} a_m x^m \qquad \ldots \qquad \textbf{\textit{(II-44)}}$$

The range of values of x for which the Taylor series converges is called the *interval of convergence*. For any point x in the interval of convergence, the Taylor series of a function f converges to the value of $f(x)$.

Using the Taylor series for a function f, we can define the action of f on *operators* over C (provided the relevant Taylor series converges). For example, we would define the exponential function so that, for an operator T, we have

$$e^T = \sum_m \frac{1}{m!} T^m \qquad \ldots \qquad \textbf{\textit{(II-45)}}$$

In general, the Taylor series for any function f acting on an operator T will have the form

$$f(T) = \sum_m a_m T^m \qquad \ldots \qquad \textbf{\textit{(II-46)}}$$

If T is written in diagonal form, then the expression simplifies:

$$f(T) = \sum_m a_m T^m = \sum_m a_m \left(\sum_i T_i |T_i\rangle\langle T_i|\right)^m = \sum_m a_m \sum_i T_i^m |T_i\rangle\langle T_i| = \sum_i f(T_i)|T_i\rangle\langle T_i| \ldots$$

(II-47)

So when T is written in diagonal form, $f(T)$ is computed by applying f separately to the diagonal entries of T. In general, the procedure for computing a function f of an operator T is to first diagonalize T (the Spectral Theorem tells us we can do this for most of the operators that will be important to us), and then compute f individually on the diagonal entries.

II-3-6- Tensor products

The *tensor product* is a way of combining spaces, vectors, or operators together. Suppose H_1 and H_2 are Hilbert spaces of dimension n and m

respectively. Then the tensor product space $H_1 \otimes H_2$ is a new, larger Hilbert space of dimension $n\times m$. Suppose $\{|b_i\rangle\} i \in \{1,\ldots,n\}$ is an orthonormal basis for $H1$ and $\{|c_j\rangle\} j \in \{1,\ldots,m\}$ is an orthonormal basis for H_2. Then

$$\{|b_i\rangle \otimes |c_j\rangle\} i \in \{1,..n\}, j \in \{1,\ldots,m\} \qquad \ldots \quad \textit{(II-48)}$$

is an orthonormal basis for the space $H_1 \otimes H_2$. The tensor product of two vectors $|\psi_1\rangle$ and $|\psi_2\rangle$ from spaces H_1 and H_2, respectively, is a vector in $H_1 \otimes H_2$, and is written $|\psi_1\rangle \otimes |\psi_2\rangle$. The tensor product is characterized by the following axioms:

1. For any $c \in$ C, $|\psi_1\rangle \in H_1$, and $|\psi_2\rangle \in H_2$,

$$c(|\psi_1\rangle \otimes |\psi_2\rangle) = (c|\psi_1\rangle) \otimes (|\psi_2\rangle) = |\psi_1\rangle \otimes (c|\psi_2\rangle) \qquad \ldots \quad \textit{(II-49)}$$

2. For any $|\psi_1\rangle, |\varphi_1\rangle \in H_1$, and $|\psi_2\rangle \in H_2$,

$$(|\psi_1\rangle + |\varphi_1\rangle) \otimes |\psi_2\rangle = |\psi_1\rangle \otimes |\psi_2\rangle + |\varphi_1\rangle \otimes |\psi_2\rangle \qquad \ldots \quad \textit{(II-50)}$$

3. For any $|\psi_1\rangle \in H_1$, and $|\psi_2\rangle, |\varphi_2\rangle \in H_2$,

$$|\psi_1\rangle \otimes (|\psi_2\rangle + |\varphi_2\rangle) = |\psi_1\rangle \otimes |\psi_2\rangle + |\psi_1\rangle \otimes |\varphi_2\rangle \qquad \ldots \quad \textit{(II-51)}$$

Suppose A and B are linear operators on H_1 and H_2 respectively. Then $A \otimes B$ is the linear operator on $H_1 \otimes H_2$ defined by

$$(A \otimes B)(|\psi_1\rangle \otimes |\psi_2\rangle) \equiv A|\psi_1\rangle \otimes B|\psi_2\rangle \qquad \ldots \quad \textit{(II-52)}$$

This definition extends linearly over the elements of $H_1 \otimes H_2$:

$$(A \otimes B)\left(\sum_{ij} \lambda_{ij} |b_i\rangle \otimes |c_j\rangle\right) \equiv \sum_{ij} \lambda_{ij} A|b_i\rangle \otimes B|c_j\rangle \qquad \ldots \quad \textit{(II-53)}$$

We have presented the tensor product using the Dirac notation. In the matrix representation, this translates as follows. Suppose A is an $m\times n$ matrix and B a $p\times q$ matrix, then the *left Kronecker product* of A with B is the $mp\times nq$ matrix

$$A \otimes B = \begin{bmatrix} A_{11}B_{11}\cdots & A_{11}B_{1q}\cdots & A_{1n}B_{11}\cdots & A_{1n}B_{1q} \\ A_{11}B_{p1}\cdots & A_{11}B_{pq}\cdots & A_{1n}B_{p1}\cdots & A_{1n}B_{pq} \\ A_{m1}B_{11}\cdots & A_{m1}B_{1q}\cdots & A_{mn}B_{11}\cdots & A_{mn}B_{1q} \\ A_{m1}B_{p1}\cdots & A_{m1}B_{pq}\cdots & A_{mn}B_{p1}\cdots & A_{mn}B_{pq} \end{bmatrix} \qquad \ldots \quad \textit{(II-54)}$$

This matrix is sometimes written more compactly in 'block form' as

$$A\otimes B=\begin{bmatrix} A_{11}[B] & A_{12}[B] & \ldots\ldots & A_{1n}[B] \\ A_{21}[B] & A_{22}[B] & \ldots\ldots\ldots & A_{2n}[B] \\ \cdot & \cdot & \cdot & \cdot \\ A_{m1}[B] & A_{m2}[B] & \ldots\ldots\ldots & A_{mn}[B] \end{bmatrix} \qquad \ldots \ \textbf{\textit{(II-55)}}$$

Here, [*B*] represents the *p*×*q* submatrix *B*. Then each block entry A_{ij} [*B*] above is the matrix [*B*] multiplied by the single entry in row *i*, column *j*, of matrix *A*.

$$A\otimes B=\begin{bmatrix} A_{ij}B_{11} & A_{ij}B_{12} & \ldots\ldots & A_{ij}B_{1q} \\ A_{ij}B_{21} & A_{ij}B_{22} & \ldots\ldots\ldots & A_{ij}B_{2q} \\ \cdot & \cdot & \cdot & \cdot \\ A_{ij}B_{p1} & A_{ij}B_{p2} & \ldots\ldots\ldots & A_{ij}B_{pq} \end{bmatrix} \qquad \ldots \ \textbf{\textit{(II-56)}}$$

The matrix representation for the tensor product of two vectors, or two operators, is the left Kronecker product of the matrix representation of the two vectors or operators being 'tensored' together. For example, the matrix representation of $(\alpha_0|0\rangle+\alpha_1|1\rangle)\otimes(\beta_0|0\rangle+\beta_1|1\rangle)$ is

$$\begin{pmatrix}\alpha_0\\ \alpha_1\end{pmatrix}\otimes\begin{pmatrix}\beta_0\\ \beta_1\end{pmatrix}=\begin{pmatrix}\alpha_0\beta_0\\ \alpha_0\beta_1\\ \alpha_1\beta_0\\ \alpha_1\beta_1\end{pmatrix} \qquad \ldots \ \textbf{\textit{(II-57)}}$$

II-3-7- The Schmidt decomposition theorem

Theorem II-6

If $|\psi\rangle$ *is a vector in a tensor product space* $H_A\otimes H_B$*, then there exists an orthonormal basis* $\{|\varphi_i^A\rangle\}$ *for HA, and an orthonormal basis* $\{|\varphi_i^B\rangle\}$ (for *HB, and non-negative real numbers* $\{p_i\}$ *so that*

$$|\psi\rangle=\sum_i\sqrt{p_i}\,|\varphi_i^A\rangle|\varphi_i^B\rangle \qquad \ldots \ \textbf{\textit{(II-58)}}$$

The coefficients $\sqrt{p_i}$ are called *Schmidt coefficients*. To understand what this theorem is saying, suppose $\{|\varphi_i^A\rangle\}$ and $\{|\varphi_i^B\rangle\}$ were chosen to be any arbitrary orthonormal bases for H_A and H_B respectively. Then, the basis states for the space $H_A\otimes H_B$ are $|\varphi_i^A\rangle\otimes|\varphi_j^B\rangle$ (often written $|\varphi_i^A\rangle|\varphi_j^B\rangle$).

The general vector $|\psi\rangle$ in $H_A\otimes\ H_B$ is then

$$\left|\psi\right\rangle = \sum_{i,j} \alpha_{i,j} \left|\varphi_i^A\right\rangle \left|\varphi_j^B\right\rangle \qquad \text{... } \textbf{(II-59)}$$

where the coefficients $\alpha_{i,j} = e^{i\phi_{i,j}} \sqrt{p_{i,j}}$ are in general complex numbers. Note that we have had to use different indices on the two sets of basis vectors to account for all the 'cross-terms'. If H_A has dimension *m* and H_B has dimension *n*, this general vector is a superposition of *mn* basis vectors. The Schmidt decomposition tells us that we can always find *some* pair of bases $\left\{\left|\varphi_i^A\right\rangle\right\}$and $\left\{\left|\varphi_j^B\right\rangle\right\}$such that all the 'cross terms' vanish, and the general vector simplifies to a sum over one set of indices

$$\left|\psi\right\rangle = \sum_{i} \sqrt{p_i} \left|\varphi_i^A\right\rangle \left|\varphi_i^B\right\rangle \qquad \text{... } \textbf{(II-60)}$$

and the coefficients can be assumed to be real (since any phase factors can be absorbed into the definitions of the basis elements). The number of terms in this sum will be (at most) the minimum of *m* and *n*.

II-3-8- Mixed States

In the preceding, we have always assumed that the state of a system has a definite state vector. Such a state is commonly referred to as a *pure* state. There are important situations, for which the qubit is described by one of a specific *set* of state vectors, with corresponding probabilities (the probabilities must add to 1). For example, suppose we know that a qubit is in the pure state $\left|\psi_1\right\rangle = \frac{1}{\sqrt{2}}\left|0\right\rangle + \frac{1}{\sqrt{2}}\left|1\right\rangle$ with probability 1/3, and is in the pure state $\left|\psi_2\right\rangle = \frac{1}{\sqrt{2}}\left|0\right\rangle - \frac{1}{\sqrt{2}}\left|1\right\rangle$ with probability 2/3. The state described by this probability distribution is called a *mixture* or *ensemble* of the states $\left|\psi_1\right\rangle$ and $\left|\psi_2\right\rangle$. We refer to the state of a system in such a situation as being a *mixed state*.

III-3-8-1- Mixed states

We can have mixed states on an ensemble of any number *n* of qubits. One way of representing a general mixed state on *n* qubits is as the ensemble

$$\{(|\psi_1\rangle, p_1), (|\psi_2\rangle, p_2), \ldots, (|\psi_k\rangle, p_k)\} \qquad \ldots \quad \textbf{\textit{(II-61)}}$$

which means that the system is in the pure (*n*-qubit) state $|\psi_i\rangle$ with probability p_i, for i = 1, 2, . . . , k. Note that a pure state can be seen as a special case of a mixed state, when all but one of the p_i equal zero.

To use a representation such as ***(II-61)*** in all our calculations would be quite cumbersome. There is an alternative, very useful, representation of mixed states in terms of operators on the Hilbert space *H*. These are called *density operators*. The matrix representation of a density operator is called a *density matrix*.

The density operator for a pure state $|\psi\rangle$ is defined as

$$\rho = |\psi\rangle\langle\psi| \qquad \ldots \quad \textbf{\textit{(II-62)}}$$

If we apply the unitary operator *U* to state $|\psi\rangle$ we get the state $U|\psi\rangle$ which has density operator $U|\psi\rangle\langle\psi|U^\dagger$. Consider measuring the state with density operator $\rho = |\psi\rangle\langle\psi|$ in the computational basis. The probability of getting 0 is given by $\langle 0|\psi\rangle\langle\psi|0\rangle = \langle 0|\rho|0\rangle$

Notice that $\langle 0|\psi\rangle\langle\psi|0\rangle$ evaluates to a real number. Since any number is the trace of a corresponding 1 × 1 matrix (whose only entry is that complex number), we can also write the probability of the measurement giving result 0 as

$$\langle 0|\psi\rangle\langle\psi|0\rangle = Tr(\langle 0|\psi\rangle\langle\psi|0\rangle) = Tr(|0\rangle\langle 0||\psi\rangle\langle\psi|) \quad \ldots \quad \textbf{\textit{(II-63)}}$$

where the last step follows from the cyclicity of trace (i.e. Tr(*ABC*) = Tr(*BCA*) = Tr(*CAB*)).

Similarly, if we measure a qubit in a state with density operator $\rho = |\psi\rangle\langle\psi|$, the probability of obtaining the outcome $|1\rangle$ is $Tr(|1\rangle\langle 1||\psi\rangle\langle\psi|)$. If only dealing with pure states, this notation is unnecessarily redundant; however, if we also consider mixed states it is a much more concise notation than used above in Equation ***(II-61).***

The density operator for an ensemble of pure states such as ***(II-61)*** is

$$\rho = \sum_{i=1}^{k} p_i |\psi_i\rangle\langle\psi_i| \qquad \ldots \quad \textbf{\textit{(II-64)}}$$

and captures all the relevant information about the state of the system.

II-3-8-2- Mixed states and the Bloch sphere

The pure states of a qubit can be represented by points on the surface of the Bloch sphere. Mixed states correspond to points in the interior of the Bloch sphere, which can be seen as follows. If $\rho = \sum_i p_i |\psi_i\rangle\langle\psi_i|$ and if the Bloch vector for $|\psi_i\rangle$ is ($\alpha_{x,i}$, $\alpha_{y,i}$, $\alpha_{z,i}$), then the Bloch vector for the mixed state ρ is

$$\rho = \sum_i p_i \left(\alpha_{x,i}, \alpha_{y,i}, \alpha_{z,i}\right) = \left(\sum_i p_i\alpha_{x,i}, \sum_i p_i\alpha_{y,i}, \sum_i p_i\alpha_{z,i}\right) \quad \ldots \quad \textbf{\textit{(II-65)}}$$

There are of course many different convex combinations of points on the surface of the Bloch sphere that correspond to the same mixed state. One can compute the Bloch vector for a mixed state directly from its density matrix as follows. If we observed that any operator on a single qubit can be written as a linear combination of operators from *{I,X, Y,Z}.*

The operators *X*, *Y,Z* all have trace 0. Since a density matrix must have trace 1, this means that any density operator for a single qubit can be written as

$$\rho = \frac{1}{2} I + \alpha_X X + \alpha_y Y + \alpha_z Z$$

The vector (αx, αy, αz) gives the coordinates for the point in the Bloch sphere corresponding to the state ρ. For example, the totally mixed state (the ensemble $\left\{\left(|0\rangle\langle 0|, \frac{1}{2}\right), \left(|1\rangle\langle 1|, \frac{1}{2}\right)\right\}$ corresponds to the point at the centre of the Bloch sphere.

II-3-9- Time-evolution of a closed system

A physical system changes in time, and so the state vector $|\psi\rangle$ of a system will actually be a function of time, $|\psi(t)\rangle$. Quantum theory postulates that the evolution of the state vector of a closed quantum system is linear. In other words, if we know that some fixed transformation, let us call it *U*, maps $|\psi_i\rangle$ to $U|\psi_i\rangle$ then

$$U\sum_i \alpha_i|\psi_i\rangle = \sum_i \alpha_i U|\psi_i\rangle \qquad \ldots \quad \textbf{(II-66)}$$

Thus, the time-evolution of the state of a *closed* quantum system is described by a unitary operator. That is, for any evolution of the closed system there exists a unitary operator *U* such that if the initial state of the system is $|\psi_1\rangle$, then after the evolution the state of the system will be

$$|\psi_2\rangle = U|\psi_1\rangle \qquad \ldots \quad \textbf{(II-67)}$$

In quantum computing, we refer to a unitary operator *U* acting on a single-qubit as a *1-qubit (unitary) gate*. We can represent operators on the 2-dimensional Hilbert space of a single qubit as 2 × 2 matrices. A linear operator is specified completely by its action on a basis.

In the principle of quantum mechanicsthe continuous timeevolution of a closed quantum mechanical system (ignoring special relativity) follows the *Schr¨odinger equation*

$$i\hbar\frac{d|\psi(t)\rangle}{dt} = H(t)|\psi(t)\rangle \qquad \ldots \quad \textbf{(II-68)}$$

where h is a physical constant known as *Planck's constant* and *H*(*t*) is a Hermitean operator known as the *Hamiltonian* of the system. The Hamiltonian is an operator which represents the total energy function for the system. It may in general be a function of time, but for convenience, let us consider Hamiltonians that are constant. In this case the solution to the Schr¨odinger equation for fixed times t_1 and t_2 is

$$|\psi(t_2)\rangle = e^{-i\hbar H(t_2-t_1)}|\psi(t_2)\rangle \qquad \ldots \quad \textbf{(II-69)}$$

For Hermitean operators *H*, the operator $e^{-iH(t2-t1)}$ is a unitary operator. So for the case of (non-relativistic and continuous time) constant Hamiltonians, one can observe that the Evolution Postulate follows from the Schr¨odinger equation.

II-3-10- Composite systems

So far we have discussed the postulates for the case of a single system only, in particular a qubit. If all we ever needed to know was how isolated qubits behave when they are never allowed to interact with each other, then this would be sufficient. If we want to study potentially useful quantum computations we will need to understand how quantum mechanics works for systems composed of several qubits *interacting* with each other. That is, we would like to know how to describe the state of a closed system of *n* qubits, how such a state evolves in time, and what happens when we measure it. Treating a larger system as a composition of subsystems (of bounded size) allows for an exponentially more efficient description of operations acting on a small number of subsystems.

Thus, when two physical systems are treated as one combined system, the state space of the combined physical system is the tensor product space $H_1 \otimes H_2$ of the state spaces H_1,H_2 of the component subsystems. If the first system is in the state $|\psi_1\rangle$and the second system in the state $|\psi_2\rangle$then the state of the combined system is

$$|\psi_1\rangle \otimes |\psi_2\rangle \qquad \ldots \quad \textit{(II-70)}$$

It is important to note that the state of a 2-qubit composite system cannot always be written in the product form $|\psi_1\rangle \otimes |\psi_2\rangle$. If the 2 qubits are prepared independently, and kept isolated, then each qubit forms a closed system, and the state *can* be written in the product form. However, if the qubits are allowed to interact, then the closed system includes both qubits together, and it may *not* be possible to write the state in the product form. When this is the case, we say that the qubits are *entangled*. From an algebraic point of view, the state of the composite system is a vector in the 4-dimensional tensor-product space of the 2 constituent qubits. The 4-dimensional state vectors that are formed by taking the tensor product of two 2-dimension state vectors form a sparse subset of all the 4-dimensional state vectors. In this sense, 'most' 2-qubit states are entangled.

II-4- Qubits

II-4-1- The state of a quantum system

A photon that is constrained to follow one of two distinguishable paths. We identified the two distinguishable paths with the 2-dimensional basis vectors $\begin{pmatrix}1\\0\end{pmatrix}and\begin{pmatrix}0\\1\end{pmatrix}$ and then noted that a general 'path state' of the photon can be described by a complex vector

$$\begin{pmatrix}\alpha_0\\\alpha_1\end{pmatrix} \qquad \dots \;\; \textbf{\textit{(II-71)}}$$

with $|\alpha_0|^2 + |\alpha_1|^2 = 1$. This simple example captures the essence of the first postulate, which tells us how physical states are represented in quantum mechanics which said that The state of a system is described by a unit vector in a Hilbert space *H*.

Depending on the degree of freedom (i.e. the type of state) of the system being considered, *H* may be infinite-dimensional. For example, if the state refers to the position of a particle that is free to occupy any point in some region of space, the associated Hilbert space is usually taken to be a continuous (and thus infinite dimensional) space. It is worth noting that in practice, with finite resources, we cannot distinguish a continuous state space from one with a discrete state space having a sufficiently small minimum spacing between adjacent locations. For describing realistic models of quantum computation, we will typically only be interested in degrees of freedom for which the state is described by a vector in a finite-dimensional (complex) Hilbert space. In particular, we will primarily be interested in composite systems composed of individual *two-level* systems. The state of each two-level system is described by a vector in a 2-dimensional Hilbert space. We can encode a *qubit* in such a two-level system. We would choose a basis for the corresponding 2-dimensional space. We would label one of the basis vectors with $|0\rangle$ and the other basis vector with$|1\rangle$. This is analogous to what is done for classical computation. For a classical computer, the two-level system may be the voltage level on a wire, which could be zero, or some positive value (say +5 mV). We might encode a classical bit in such a system by assigning the binary value ‘0’ to the state in which the voltage on the wire is 0, and the value ‘1’ to the state in which the voltage on the wire is + 5 mV. The $\{|0\rangle, |1\rangle\}$ basis for the state of a qubit is commonly referred to as the *computational basis*.

The state of this system is described by a vector in a 2-dimensional Hilbert space. A convenient basis for this space consists of a unit vector for the state in which a photon is not present, and an orthogonal unit vector for the state in which a photon is present. We can label these states with $|0\rangle \text{and} |1\rangle$, respectively. Then the general state of the system is expressed by the vector

$\alpha_0|0\rangle + \alpha_1|1\rangle$

where α_0 and α_1 are complex coefficients, often called the *amplitudes* of the basis states $|0\rangle \text{and} |1\rangle$, respectively. Note that a complex amplitude α can be decomposed unique as a product $e^{i\theta}|\alpha|$ where $|\alpha|$ is the non-negative real number corresponding to the magnitude of α, and $e^{i\theta} = \frac{\alpha}{|\alpha|}$ has norm 1. The value θ is known as the ‘phase’, and we refer to the value $e^{i\theta}$ as a ‘phase factor’.

The condition that the state is described by a *unit* vector means that $|\alpha_0|^2 + |\alpha_1|^2 = 1$. This condition is sometimes called the *normalization constraint*, and it is necessary for consistency with the way quantum measurements behave. The general state of the system is a superposition of a photon being present, and a photon not being present.

Another example of a two-level quantum mechanical system is the spin state of certain types of particles. According to quantum physics, particles have a degree of freedom called *spin*, which does not exist in a classical description. Many particles fall into the category of so called *spin*1/2particles. For these, the *spin state* is indeed described by a vector in a 2-dimensional Hilbert space *H*. A convenient basis for this space consists of a unit vector for the ‘spin-up’ state of the particle, and an orthogonal unit vector for the ‘spin-down’ state of the particle. We can label these basis vectors by $|0\rangle \text{and} |1\rangle$, respectively. The general *spin state* of a spin1/2 particle is a superposition of spin-up and spin-down.

An important point about state vectors is the following. The state described by the vector $e^{i\theta}|\psi\rangle$ is equivalent to the state described by the vector $|\psi\rangle$, where $e^{i\theta}$ is any complex number of unit norm. For example, the state $|0\rangle+|1\rangle$ is equivalent to the state described by the vector $e^{i\theta}|0\rangle+e^{i\theta}|1\rangle$.

On the other hand, *relative phase factors* between two orthogonal states in superposition *are* physically significant, and the state described by the vector $|0\rangle+|1\rangle$ is physically different from the state described by the vector $|0\rangle+e^{i\theta}|1\rangle$

So the State Space Postulate, together with the observation of the previous paragraph, tells us that we can describe the most general state $|\psi\rangle$ of a single qubit by a vector of the form

$$|\psi\rangle=\cos\left(\frac{\theta}{2}\right)|0\rangle+e^{i\varphi}\sin\left(\frac{\theta}{2}\right)|1\rangle \qquad \dots \textbf{\textit{(II-72)}}$$

Consider the analogous situation for a deterministic classical bit. The state of a classical bit can be described by a single binary value ψ, which can be equal to either 0 or 1***(Figure II-3)***

Figure II-3: The state of a deterministic classical bit can be represented as one of two points, labelled ‘0’ and ‘1’.

In this figure, the state can be indicated by a point in one of two positions, indicated by the two points labelled 0 and 1.

Next consider the slightly more complicated situation of a classical bit whose value is not known exactly, but is known to be either 0 or 1 with corresponding probabilities p_0 and p_1. We might call this a *probabilistic classical bit*. The state of such a probabilistic bit is described by the probabilities *p*0 and *p*1, which satisfy $p_0 + p_1 = 1$. We can represent these two probabilities by the 2-dimensional unit vector $\begin{pmatrix} p_0 \\ p_1 \end{pmatrix}$ whose entries are restricted to be real and non-negative ***(Figure II-4).*** In this figure, the state could be drawn as a point on the line between the positions 0 and 1. We suppose this line has unit length, and the position of the point on the line is determined by the probabilities p_0 and p_1.

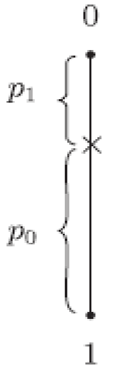


Figure II-4: A probabilistic classical bit. Here the probabilities p_0 and p_1 of the bit being 0 and 1, respectively, are represented by the position of a point on the line segment between the points representing 0 and 1.

Note that with only one copy of such a probabilistic bit, we cannot determine p_0 and p_1 exactly. *If* we are given a means to obtain several independent copies of the probabilistic bit then we could accumulate statistics about the values p_0 and p_1. Otherwise, we cannot in general ‘clone’ this bit and get two or more independent copies that would allow us to obtain arbitrarily good estimates of p_0 and p_1.

Now return to the state of a quantum bit, which is described by a complex unit vector $|\psi\rangle$ in a 2-dimensional Hilbert space. Up to a global phase factor, such a vector can always be written in the form

$$|\psi\rangle = \cos\left(\frac{\theta}{2}\right)|0\rangle + e^{i\varphi}\sin\left(\frac{\theta}{2}\right)|1\rangle \qquad \ldots \quad \textbf{\textit{(II-73)}}$$

Such a state vector is often depicted as a point on the surface of a 3-dimensional sphere, known as the *Bloch sphere* ***(Figure II-5).*** Two real parameters θ and φ are sufficient to describe a state vector, since state vectors are constrained to have norm 1 and are equivalent up to global phase. Points on the surface of the Bloch sphere can also be expressed in Cartesian coordinates as

$$(x, y, z) = (\sin\theta\cos\varphi, \sin\theta\sin\varphi, \cos\theta) \qquad \dots \textit{(II-74)}$$

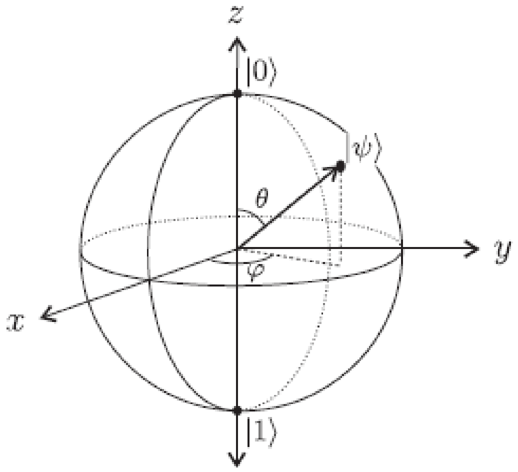


Figure II-5: State of a qubit on the Bloch sphere.

II-4-2- General Quantum Operations

A superoperator or a ‘general quantum operation’ can take as input a system described by a density operator ρ_{in} corresponding to a Hilbert space of dimension *N*, add an ancilla of arbitrary size (in fact, it can be shown, using Caratheodory’s Theorem, that the dimension of the ancilla never needs to be larger than N^2 and that we can assume without loss of generality that the ancilla is initialized to some fixed pure state), perform a unitary operation *U* on the joint system, and then discard some subsystem.

More explicitly, this can be described as the map:

$$\rho_{in} \mapsto \rho_{out} = Tr_B\left(U\left(\rho_{in} \otimes |00..0\rangle\langle 00...0| U^{+}\right)\right) \qquad \dots \textit{(II-75)}$$

where the state $|00..0\rangle$ is an ancilla state of arbitrary size (but without loss of generality has dimension at most N^2), *U* is a unitary operation acting on the joint system, and *B* is some subsystem of the joint system.

If *B* is the original ancilla system, then the superoperator does not change the Hilbert space of the system. In general, we can describe states that change the dimension of the state space. It is shown that the action of such a superoperator (restricting attention to operators that do not change the Hilbert space) can be described by a finite sum

$$\rho_{in} \mapsto \sum_i A_i \rho_{in} A_i^+ \qquad \ldots \ \textbf{\textit{(II-76)}}$$

where the A_i are called *Kraus operators*, which are linear operators9 on the same Hilbert space as ρ_{in} and satisfy

$$\sum_i A_i A_i^+ = I \qquad \ldots \ \textbf{\textit{(II-77)}}$$

Conversely, every set of Kraus operators satisfying the completeness condition ***(Equation II-76)*** can be realized by a map of the form in Equation ***(II-75)*** for some unitary *U* (which is unique up to a final unitary on the system that is traced out).

II-4-3- The Hadamard transformation

One of the most basic transformations in quantum information science is the so-called Hadamard transformation whose actions on a qubit are

$$H|0\rangle \rightarrow \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = |Q^{'}\rangle, \ \ H|1\rangle \rightarrow \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \ \ \ldots \qquad \textbf{\textit{(II-78)}}$$

Applying this to the qubit lQ'> above, results in

$$H|Q^{'}\rangle = |0\rangle \qquad \ldots \quad \textbf{\textit{(II-79)}}$$

That is, a well defined value of the qubit. This is never possible with an incoherent mixture.

II-4-4- Single Qubit Transformations:

Insight in some of the most basic experimental procedures in quantum information physics can be gained by investigating the action of a simple 50/50 beamsplitter. Such beamsplitters have been realized for many different types of particles not only for photons. For a general beamsplitters, as shown in ***Figure II-6***, let us investigate the case of just two incoming modes and two outgoing modes which are arranged as shown in the figure.

For a. 50/50 beamsplitter a particle incident either from above or from below has the same probability of 50% of emerging in either output beam, above or below. Then quantum unitarily, that is, the requirement that no particles are lost if the beamsplitter is non-absorbing, implies certain phase conditions on the action of the beamsplitter with one free phase. A very simple way to describe the action of a beamsplitter is to fix the phase relations such that the beamsplitter is described by the Hadamard transformation of the equation ***(II-78)***.

Lets us again assume that the incident state s the general qubit.

$$\left|Q\right\rangle_{in} = \alpha\left|0\right\rangle_{in} + \beta\left|1\right\rangle_{in} \qquad \dots \textbf{\textit{(II-80)}}$$

For a single incident particle this means that α is the probability amplitude to find the particle incident from above and β is the probability amplitude for finding the particle incident from below. Then the action of the beamsplitter results in the final state.

$$\left|Q\right\rangle_{out} = H\left|Q\right\rangle_{in} = \frac{1}{\sqrt{2}}((a+B)\left|0\right\rangle_{out} + (a-\beta)\left|1\right\rangle_{out}) \quad \dots \textbf{\textit{(II-81)}}$$

Where (α + β) is now the probability amplitude for finding in the particle the outgoing upper bean and (α - β) is the probability amplitude for finding it in the outgoing lower bean. For the specific case of α=0 or β=0, we find that the particle will be found with equal probability in either of the outgoing beams. For another specific case, α = β, we find that the particle will definitely be found in the upper beam and never in the lower beam.

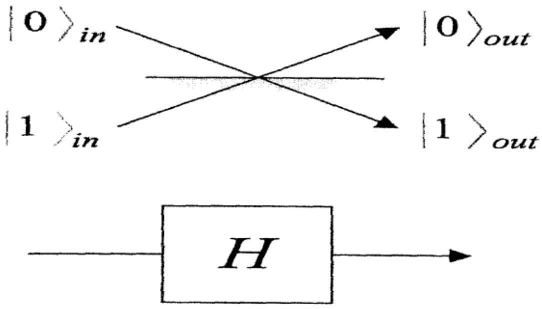


Figure II-6: The 50/50 beamspliter (top) and the corresponding diagram using the Hadamard transform (below) [24].

It is interesting and instructive to consider sequences of such beamsplitters because they realise sequences of Hadamard transformations.

Furthermore, the mirrors shown only serve to redirect the beams, they are assumed to have identical action on the two beams and therefore can be omitted in the analysis. The full action of the interferometer can now simply be described as two successive Hadamard transformations acting on the general incoming state of the equation ***(II-80)***.

$$|Q\rangle_{out} = HH|Q\rangle_{in} = |Q\rangle_{in} \qquad \dots \qquad \textbf{\textit{(II-82)}}$$

This result from the simple fact double application of the Hadamard transformation of ***(II-78)*** is the identity operation. It means that the Mach Zehnder interferometer as sketched in ***Figure II-7***, with beamsplitters realizing the Hadamard transformation at its output reproduces a state identical to the input. Let us consider again the extreme case where the input consists of one beam only, that is, without loss of generality, let us assume a=1, the lower beams being empty then, according to the equation ***(II-82)*** the particle will definitely be found in the upper output. Most interestingly, this is because between the two beamsplitters the particle would have been found (with the correct relative phase) with equal probability in both beam paths. It is the interference of the two amplitudes incident on the final beamsplitter which results in the particle ending up with certainty in one of outgoing beams and never in the other.

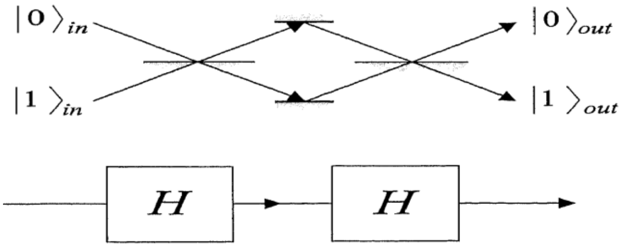


Figure II-7: A Mach-Zehner interferometer (top) is a sequence of two Hadamard transformations (bottom) [24].

In quantum information language, the output qubit of the empty MachZender interferometer will have a definite value if the input qubit also has only because between the two Hadamard transformations the value of the qubit was maximally underlined.

Another important quantum gate besides the hadamard gate is the phase shifter, which is introduced additionally in ***Figure II-8*** into the Mach- Zehnder interferometer. Its operation is simple to introduce a phase change φ to the amplitude of one of the two beams (without loss of generality we can assume this to be the upper beam because only relative phases are relevant). In our notation., the action of the phase shifter can be described by the unitary transformation.

$$\phi|0\rangle = e^{i\varphi}|0\rangle, \ \phi|1\rangle = |1\rangle, \qquad \dots \ \textbf{\textit{(II-83)}}$$

Therefore the output can be calculated by successive application of all proper transformations to the input qubit:

$$|Q\rangle_{out} = H\phi H|Q\rangle_{in} \qquad \dots \ \textbf{\textit{(II-84)}}$$

We will restrict our discussion again to the case where we have only one input namely $\alpha = 1$. And $\beta = 0$, i.e., $|Q\rangle_{in} = 0$. The final state then become

$$H\phi H|0\rangle = \frac{1}{2}\left(\left(e^{i\varphi}+1\right)|0\rangle + \left(e^{i\varphi}-1\right)|1\rangle\right) \qquad \dots \ \textbf{\textit{(II-85)}}$$

This has a very simple interpretation. First we observe by inspection of the equation ***(II-74)*** that for $\varphi = 0$ the value of the qubit is definitely “0”. On the other hand, for $\varphi = \pi$ the value of the qubit is definitely “1”. This indicates that the phase shift φ is able to switch the output qubit has the value “0” is $P_0 = \cos^2(\varphi/2)$, and the probability that the qubit carries the value “1” is $P_1 = \sin^2(\varphi/2)$.

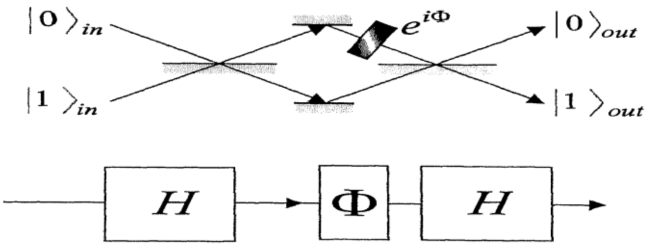


Figure II-8 : Top : Mach-Zehnder interferometer including a phase shifter ϕ in one of the two beams. Bottom: The equivalent representation with Hadamard transformations and a phase shifter gate.

III-1- Introduction:

Technological growth in the electronics industry has historically been measured by the number of transistors that can be crammed onto a single microchip. Unfortunately, all good things must come to an end; spectacular growth in the number of transistors on a chip requires spectacular reduction of the transistor size. For electrons in semiconductors, the laws of quantum mechanics take over at the nanometer scale, and the conventional wisdom for progress must be abandoned. This realization has stimulated extensive research on ways to exploit the spin (in addition to the orbital) degree of freedom of the electron, giving birth to the field of spintronics. Perhaps the most ambitious goal of spintronics is to realize complete control over the quantum mechanical nature of the relevant spins. This prospect has motivated a race to design and build a spintronics device capable of complete control over its quantum mechanical state, and ultimately, performing computations: a quantum computer.

III-2- Electron Spin

III-2-1- Toward the world of spin

Two types of experimental evidence which arose in the 1920s suggested an additional property of the electron. One was the closely spaced splitting of the hydrogen spectral lines, called fine structure. The other was the Stern-Gerlach experiment which showed in 1922 that a beam of silver atoms directed through an inhomogeneous magnetic field would be forced into two beams [5]. Both of these experimental situations were consistent with the possession of an intrinsic angular momentum and a magnetic moment by individual electrons. Classically this could occur if the electron was a spinning ball of charge, and this property was called electron spin.

Quantization of angular momentum had already arisen for orbital angular momentum, and if this electron spin behaved the same way, an angular momentum quantum number s = 1/2 was required to give just two states. This intrinsic electron property gives:

Z-component of angular momentum: $S_z = m_s\hbar \quad ,m_s = \pm\frac{1}{2}\ldots$ ***(III-1)***

Magnetic moment: $\mu_s = -\frac{e}{2m}gS\ \ldots$ ***(III-2)***

An electron spin s = 1/2 is an intrinsic property of electrons. Electrons have intrinsic angular momentum characterized by quantum number 1/2. In the pattern of other quantized angular momenta, this gives total angular momentum

$$S = \sqrt{\frac{1}{2}\left(\frac{1}{2}+1\right)}\hbar = \frac{\sqrt{3}}{2}\hbar \qquad \ldots \qquad \textbf{\textit{(III-3)}}$$

The resulting fine structure which is observed corresponds to two possibilities for the z-component of the angular momentum.

$$S_z = \pm\frac{1}{2}\hbar \qquad \ldots \qquad \textbf{\textit{(III-4)}}$$

This causes an energy splitting because of the magnetic moment of the electron

$$\mu_s = -\frac{e}{2m}gS \qquad \ldots \qquad \textbf{\textit{(III-5)}}$$

III-2-2- Electron Intrinsic Angular Momentum

Experimental evidence like the Stern-Gerlach experiment suggests that an electron has an intrinsic angular momentum, independent of its orbital angular momentum [5]. These experiments suggest just two possible states for this angular momentum, and following the pattern of quantized angular momentum, this requires an angular momentum quantum number of 1/2.

With this evidence, we say that the electron has spin ½. An angular momentum and a magnetic moment could indeed arise from a spinning sphere of charge, but this classical picture cannot fit the size or quantized nature of the electron spin. The property called electron spin must be considered to be a quantum concept without detailed classical analogy. The quantum numbers associated with electron spin follow the characteristic pattern:

$$S = \sqrt{S(S+1)}\hbar, \quad S = \frac{1}{2}, \quad ,m_s = \pm\frac{1}{2} \quad \ldots \qquad \textbf{\textit{(III-6)}}$$

III-2-3- Electron Spin Magnetic Moment

Since the electron displays an intrinsic angular momentum, one might expect a magnetic moment which follows the form of that for an electron orbit. The z-component of magnetic moment associated with the electron spin would then be expected to be $\mu_z = \pm\frac{1}{2}\mu_B$

but the measured value turns out to be about twice that. The measured value is written

$$\mu_z = \pm\frac{1}{2}g\mu_B \quad \ldots \qquad \textbf{(III-7)}$$

Where g is called the gyromagnetic ratio and the electron spin g-factor has the value g = 2.00232 and g=1 for orbital angular momentum. A natural constant which arises in the treatment of magnetic effects is called the Bohr magneton. The magnetic moment is usually expressed as a multiple of the Bohr magneton.

$$\mu_B = \frac{e\hbar}{2m_e} = 9.2740154*10^{-24}\,J/T = 5.7883826*10^{-5}\,ev/T \Rightarrow Bohr \quad magneton\ldots \qquad \textbf{(III-8)}$$

The electron spin magnetic moment is important in the spin-orbit interaction which splits atomic energy levels and gives rise to fine structure in the spectra of atoms. The electron spin magnetic moment is also a factor in the interaction of atoms with external magnetic fields (Zeeman Effect).

The term "electron spin" is not to be taken literally in the classical sense as a description of the origin of the magnetic moment described above. To be sure, a spinning sphere of charge can produce a magnetic moment, but the magnitude of the magnetic moment obtained above cannot be reasonably modeled by considering the electron as a spinning sphere. High energy scattering from electrons shows no "size" of the electron down to a resolution of about 10^{-3} fermi, and at that size a preposterously high spin rate of some 10^{32} radian/s would be required to match the observed angular momentum.

III-2-3-1- Zeeman Interaction and Zeeman Effect

An external magnetic field will exert a torque on a magnetic dipole and the magnetic potential energy which results is

$$u(\theta) = -\mu * B \quad \ldots \qquad \textbf{(III-9)}$$

The magnetic dipole moment associated with the orbital angular momentum is given by

$$\mu_{orbital} = -\frac{2}{2m_e}L \qquad \ldots \qquad \textbf{(III-10)}$$

For magnetic field in the z-direction this gives

$$U = \frac{e}{2m}L_z B = m_l \frac{e\hbar}{2m}B \qquad \ldots \qquad \textbf{(III-11)}$$

Considering the quantization of angular momentum, this gives equally spaced energy levels displaced from the zero field level by

$$\Delta E = m_l \frac{e\hbar}{2m}B = m_l \mu_B B \qquad \ldots \qquad \textbf{(III-12)}$$

This displacement of the energy levels gives the uniformly spaced multiplet splitting of the spectral lines which is called the Zeeman Effect.

The magnetic field also interacts with the electron spin magnetic moment, so it contributes to the Zeeman effect in many cases. The electron spin had not been discovered at the time of Zeeman's original experiments, so the cases where it contributed were considered to be anomalous. The term "anomalous Zeeman effect" has persisted for the cases where spin contributes. In general, both orbital and spin moments are involved, and the Zeeman interaction takes the form

$$\Delta E = \frac{e}{2m}\left(\vec{L} + 2\vec{S}\right).\vec{B} = g_L m_j \mu_B B \qquad \ldots \qquad \textbf{(III-13)}$$

The factor of two multiplying the electron spin angular momentum comes from the fact that it is twice as effective in producing magnetic moment. This factor is called the spin g-factor or gyromagnetic ratio. The evaluation of the scalar product between the angular momenta and the magnetic field here is complicated by the fact that the S and L vectors are both precessing around the magnetic field and are not in general in the same direction. The persistent early spectroscopists worked out a way to calculate the effect of the directions. The resulting geometric factor g_L in the final expression above is called the Lande g factor. It allowed them to express the resultant splittings of the spectral lines in terms of the z-component of the total angular momentum, m_j. The above treatment of the Zeeman effect describes the phenomenon when the magnetic fields are small enough that the orbital and spin angular momenta can be considered to be coupled.

III-2-3-2- The Electron Spin g-factor

When the Zeeman Effect, the observed splitting was consistent with an electron orbit magnetic moment given by

$\mu_{orbital} = -\frac{e}{2m}\vec{L}$ giving energy shift of form

$$\frac{e\hbar}{2m} m_l . B = m_j \mu_B B \quad \ldots \qquad \textbf{(III-14)}$$

where the splittings followed the z-component of angular momentum and the selection rules explained why you got a triplet of closely-spaced lines for the 3 > 2 transition of hydrogen. But when the effects of electron spin were discovered by Goudsmit and Uhlenbeck [5], they found that the observed spectral features were matched by assigning to the electron spin a magnetic moment

$$\mu_{spin} = -g\frac{e}{2m}\vec{S} \qquad \ldots \qquad \textbf{(III-15)}$$

where g is approximately 2.

III-2-4- Spin orbit interaction:

Although neglected up to this lecture, the interaction between the electron-spin and the orbital angular momentum must also be included in the atomic Hamiltonian. Such interaction is described according to the *spin-orbit Hamiltonian* defined as follows,

$$\hat{H}_{so} = \frac{1}{2m_e c^2}\frac{1}{r}\left(\frac{\delta V}{\delta r}\right)\hat{L}\hat{S} = \varepsilon\,\hat{L}.\hat{S} \quad \ldots \qquad \textbf{(III-16)}$$

Where *V* is the Coulombic potential of the electron in the field of the atom. Note that the spin-orbit interaction is proportional to $\hat{L}.\hat{S}$.

A classical description of such interaction also gives a perturbation proportional to $\hat{L}.\hat{S}$. This is because from the reference frame of the electron, the nucleus is a moving charge that generates a magnetic field *B*, proportional to $\hat{L}$. Such magnetic field interacts with the spin magnetic moment $m_s = \frac{-e}{m_e \hat{S}}$. Therefore, the interaction between *B* and m$_s$ is proportional to $\hat{L}.\hat{S}$.

III-3- General concepts in spin-transport

III-3-1- Basic transport mechanism in a magnetic device

Although these concepts are discussed in more details in [21], now let us briefly maintain some of these concepts. Let us consider the prototypical GMR (Giant magnetoresistance) device: the spin-valve. A spin valve is formed by two magnetic layers separated by a non-magnetic spacer. Usually the magnetic layers are metallic (typically Co, Ni, Fe or some permalloy), while the spacer can be either a metal, a semiconductor, an insulator or a nanoscale object such as a molecule or an atomic constriction. The typical operation of a spin-valve is schematically illustrated in ***Figure III-1***. Usually the two magnetic layers have a rather different magnetic anisotropy with one layer being strongly pinned and the other free to rotate along an external magnetic field. In this way the magnetotransport response of the device can be directly related to the direction of the magnetization of the free layer. In our discussion, we consider only the two extreme cases in which the two magnetization vectors are either parallel (P) or antiparallel (AP) to each other.

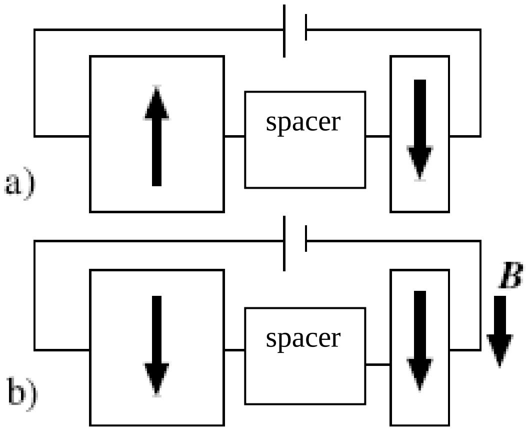


Figure III-1: Scheme of a spin-valve in the two resistance states: **a)** high resistance, **b)** low resistance. [14]

To fix the idea consider a Co/Cu/Co spin-valve, and let us follow the path of both the electron spin species across the device. The Fermi surfaces line up for both the P and AP cases are presented in ***FigureIII-2***. In the AP case the magnetization vector of the two magnetic layers points in opposite directions. This means that an electron belonging to the majority band in one layer will belong to the minority in the other layer. Consequently in the AP case both the spin currents (usually called the spin channels) arise from electrons that have travelled within the Fermi surface of Cu and of both spins of Co. In contrast in the P case the two spin currents are rather different. The spin up current is made from electrons that have travelled within the Fermi surfaces of Cu and of the majority spin Co, while the down spin current from electrons that have traveled within the Fermi surfaces of Cu and of the minority spin Co. If we naively assume that the total resistance of the device can be obtained by adding in series the resistances of the materials forming the device (resistor network model) we obtain:

$$R_{AP} = \frac{1}{2}(R_{\uparrow}^{co} + R_{\downarrow}^{co} + R^{cu}), \qquad R_P = \left(\frac{1}{2R_{\uparrow}^{co} + R^{cu}} + \frac{1}{2R_{\downarrow}^{co} + R^{cu}}\right)^{-1} \quad \ldots \quad \textbf{\textit{(III-17)}}$$

where R_P and R_{AP} are the resistance for the parallel and antiparallel configuration respectively, R^{Cu} is the resistance of the Cu layer and $R_{\uparrow}^{co}$ and $R_{\downarrow}^{co}$ are the resistance of the Co layer for the majority (↑) and the minority (↓) spins. Usually $R_{\uparrow}^{co} \prec\prec R_{\downarrow}^{co}$, hence $R_P < R_{AP}$. This produces the GMR effect (see III-4-1).

Conventionally the magnitude of the effect is given by the GMR ratio r_{GMR} defined as:

$$r_{GMR} = (R_{AP} - R_P)/R_P \ldots \qquad \textbf{\textit{(III-18)}}.$$

This is usually called the “optimistic” definition (since it gives large ratios). An alternative definition is obtained by normalizing the resistance difference by either R_{AP} or $R_P + R_{AP}$; in this last case r_{GMR} is bounded between 0 and 1.

The discussion so far is based on the hypothesis of treating the spin-valve as a resistor network. This is strictly true only if $\lambda_{emf} < l\varphi < L$, where L is the typical thickness of the layers forming the spin-valve, but in general adding resistances in series may not be correct. However it is also clear that the magnitude of the magnetoresistance depends critically on the asymmetry of the two spin currents in the magnetic material, which ultimately depends on its electronic structure. It is therefore natural to introduce the concept of spin polarization P of a magnetic metals as $P = (I_{\uparrow} - I_{\downarrow})/ (I_{\uparrow} + I_{\downarrow})$, where I_{σ} is the spin-σ contribution to the current.

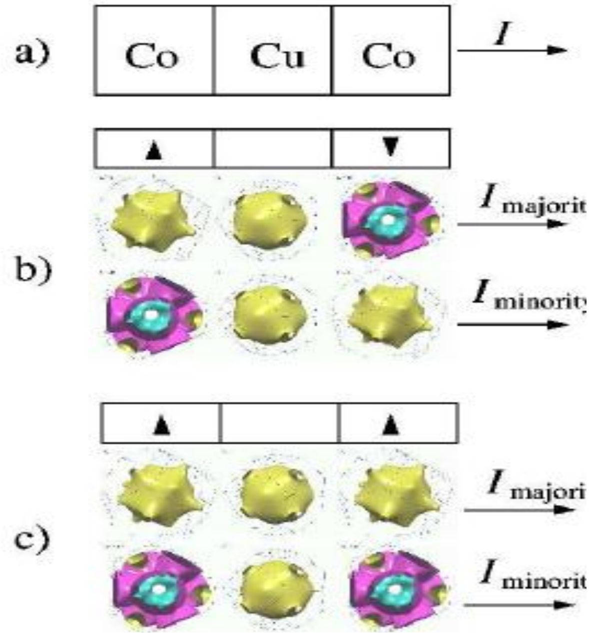


Figure III-2: Magnetoresistance mechanism in a Co/Cu/Co spin valve (panel a) in the two spin current approximation. [14]

III-3-2- Transport regimes:

the relation between the spin-polarization of a magnetic material and its electronic structure depends critically on the transport regime that one considers.

III-3-2-1- Diffusive Transport

In diffusive transport the phase coherence length is rather short and quantum interference is averaged out. The transport is then described by the Boltzmann's equations, which govern the evolution of the electron momentum distribution function. Within the relaxation time approximation, assuming that the relaxation times does not depend on the electron spin the current is simply proportional to $N_F v^2_F$, where N_F and v_F are the density of states at the Fermi level and the Fermi velocity respectively.

This leads us to the " Nv^2 " definition of the spin-polarization:

$$P_{Nv^2} = \frac{N_F^{\uparrow} v_F^{2\uparrow} - N_F^{\downarrow} v_F^{2\downarrow}}{N_F^{\uparrow} v_F^{2\uparrow} + N_F^{\downarrow} v_F^{2\downarrow}} \quad \ldots \qquad \textbf{\textit{(III-19)}}$$

III-3-2-2- Ballistic Transport

In this case lφ is much longer than the size of the magnetic device. The energy is not dissipated as resistance in the device and the current can be calculated using the Landauer formalism. Also, the current, are simply proportional to N_Fv_F. Moreover in the Landauer approach, the electron velocity and the density of states exactly cancel. This means that N_Fv_F is just an integer proportional to the number of bands crossing the Fermi level in the direction of the transport, or alternatively to the projection of the Fermi surface on the plane perpendicular to the direction of the transport.

This leads to the "Nv" definition of spin-polarization:

$$P_{Nv} = \frac{N_F^{\uparrow}v_F^{\uparrow} - N_F^{\downarrow}v_F^{\downarrow}}{N_F^{\uparrow}v_F^{\uparrow} + N_F^{\downarrow}v_F^{\downarrow}} \quad \cdots \qquad \textbf{\textit{(III-20)}}$$

III-3-2-3- Tunneling

It is generally acknowledged that in tunneling experiments the GMR ratio of the specific device is given by some density of states. This was firstly observed by Jullier almost three decades ago and it is based on the fact that typical tunneling times are much faster then L/v_F, with L being the length of the tunneling barrier. This means that the electron velocity in the metal is irrelevant in the tunneling process. Although it is now clear that the relevant density of states for magneto-tunneling processes is not necessarily that of the bulk magnetic metal, but it must take into account of the structure of the tunneling barrier and of the bonding between the barrier and the metal, we can still introduce the "N" definition of polarization:

$$P_N = \frac{N_F^{\uparrow} - N_F^{\downarrow}}{N_F^{\uparrow} + N_F^{\downarrow}} \quad \cdots \qquad \textbf{\textit{(III-21)}}$$

III-3-3- Crossover between different transport regimes

Clearly the three definitions may give rise to different spin-polarizations, since the relative weight of N and v is different. In particular P_N favours electrons with high density, while P_{Nv}^2 electrons with high mobility. In magnetic transition metals, where high mobility low density s electrons coexist with low mobility high density d electrons, these differences can be largely amplified. In principle one can speculate around materials that are normal metals according to one definition and half-metals according to another. This is for instance the case of $La_{0.7}A_{0.3}MnO_3$ with A=Ca, Sr, .., in which the majority band is dominated by delocalized example states and the minority by localized t_{2g} electrons. Therefore $La_{0.7}A_{0.3}MnO_3$ is a conventional ferromagnet according to the definitions P_{Nv} and P_N and it is an half-metal according to $P_{Nv}{}^2$.

III-3-3-1- Spin-transport at the atomic level

The main idea is now to shrink the dimensions of the device in such a way that it's sensitive part will be of a size comparable with the Fermi wave length. In this case the transport is ballistic and depends critically on the entire device. Therefore it can be hardly inferred from the properties of its components, such as the spin-polarization of the current/voltage electrodes.

Let us use again the spin-valve as a prototypical example, and consider two magnetic bulk contacts separated by an atomic scale object. This can be a point contact or for instance a molecule. There are two main differences with respect to the bulk case:

1) the Fermi surface of the spacer can be highly degenerate, in the extreme limit collapsing into a single point, 2) the coupling between the magnetic surfaces and the spacer can be strongly orbital dependent. The crucial point is that in both cases the transport characteristics will be given by local properties of the Fermi surfaces of the magnetic material, which means either from a particular region in k-space, or a particular orbital manifold.

Consider ***Figure III-3*** where we present an hypothetical device formed by two metallic surfaces sandwiching a spacer whose Fermi surface is a single point. For the sake of simplicity, we consider a model ferromagnet, namely a single orbital two-dimensional simple cubic lattice, with Fermi surfaces centred at the band center and at the band edges respectively for majority and minority spins. In this case the Fermi surface of the spacer overlaps only with the majority Fermi surface of the magnetic material. For this reason we expect zero transmission for the minority spins and for the antiparallel configuration, leading to an infinite GMR ratio.

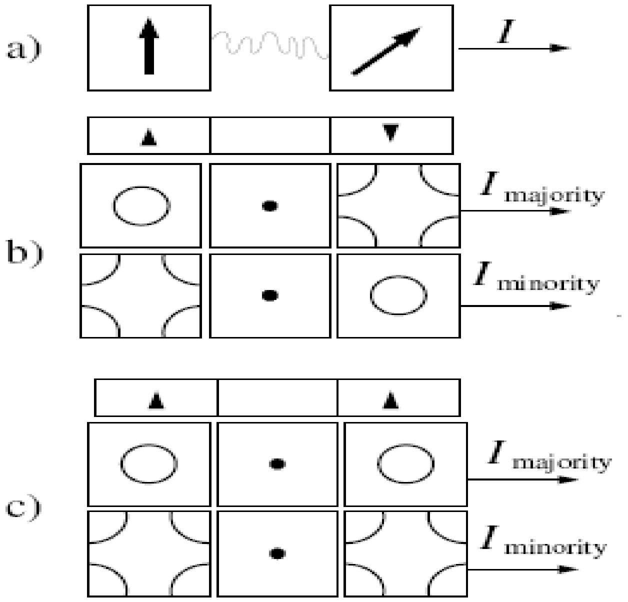


Figure III-3: Magnetoresistance mechanism in a spin valve constructed with an atomic scaled spacer in the two spin current approximation. [15]

III-3-3-2- Molecular Spin transport

The growing interest in interfacing conventional electronic devices with organic compounds has brought to the construction of spin-valves using molecules as a spacer. These include carbon nanotubes, elementary molecules and polymers. Spin-transport through these objects can be highly non-conventional and vary from metallic-like, to Coulomb-blockade like, to tunneling-like. Moreover the molecule can be either chemisorbed of physisorbed depending on the molecular end groups, and the same spacer can give rise to different transport regimes.

In this case the simple requirement of local charge neutrality is not enough to describe the physics of the spacer and an accurate description of the drop of the electrostatic potential across the device is needed. Note that the transport can still be completely ballistic, in the sense that the electrons do not change their energy while crossing the spacer.

A more complicate situation arises for polymer-like spacers. In most polymers in fact the transport is not band transport but it is due to hopping and it is associated with the formation and propagation of polarons. Clearly this adds additional complication to the problem since now the electronic and ionic degrees of freedom cannot be decoupled in the usual Born-Oppenheimer approximation. At present there is very little theoretical work on spin-transport in polymers.

III-4- Transport Theory: Linear Response

III-4-1- GMR

As already mentioned the GMR effect is the drastic change in resistance of a magnetic multilayer when a magnetic field is applied. This is related to the change of the mutual orientation of the magnetic moments of the magnetic layers. In metallic systems, adjacent magnetic layers are magnetically coupled to each other, through the non-magnetic ones. The sign of this exchange coupling, discovered by Stuart Parkin in the early nineties [17], is an oscillatory function of the separation between the magnetic layers, whose details depend on the Fermi surface of the non-magnetic one. In practice, one can tune the thickness of the non-magnetic layers to obtain an overall antiferromagnetic (AF) state of the multilayer. In this situation the multilayer is in a high resistance state. When a magnetic field strong enough to align the magnetic layer along the same direction is applied, thus overcoming the antiferromagnetic exchange coupling, the multilayer resistance drops. Now the system is in a ferromagnetic (FM) configuration corresponding to a low resistance state. The relative change in resistance is the GMR effect. Early GMR experiments have been conducted with the so-called current in the plane configuration (CIP) ***(Figure III-4)*** in which the current flows in the plane of the layers. In these experiments the typical cross sections are of the order of 1 mm^2 and the transport is mainly diffusive. This is the favourite configuration for devices, since the resistances are rather large and they can be measured with conventional four-probe technique.

An important breakthrough was the possibility to study the transport of a multilayer with the current flowing perpendicular to the planes (CPP GMR). In this case the resistances are rather small and difficult to measure, and one must either use superconducting contacts or shape the samples to very small cross sections. In these experiments the electrons have to cross the entire multilayer over distances smaller than 1 μm. The spin filtering is more effective and the transport can be phase-coherent.

The CPP arrangement is preferred by theoreticians since ab initio calculations can be carried out.

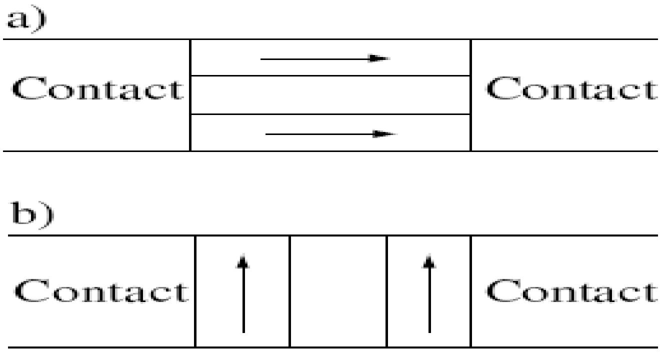


Figure III-4: Schematic representation of a typical GMR experiment: ***a)*** Current In Plane (CIP) configuration, ***b)*** Current Perpendicular to the Planes (CPP) configuration. [16]

III-4-2- TMR (Tunnelling magnetoresistance)

Despite the indisputable success of the CPP GMR either as a scientific tool or as building block for devices, it presents some disadvantages in practical applications. Firstly, since the layer thicknesses are rather small there is the need of measuring the resistance with sophisticated techniques such as superconducting contacts, which clearly are not practical for applications. Secondly it is generally difficult to magnetically decouple the layers, large magnetic fields are needed and complex micromagnetic effects are unavoidable. In order to overcome these difficulties a much simpler structure has been proposed. This is a tunneling junction, formed by two magnetic layers sandwiching an insulating material and connected to two current/voltage probes. The two layers are now magnetically decoupled and engineered to have different coercive fields; hence their mutual orientation can be changed by applying a tiny magnetic field. Also in this case the high current state is the ferromagnetic and the low current state the antiferromagnetic. The quality of the device is measured by the tunneling magnetoresistance ratio (TMR) using the same definition of that for GMR.

The main difference between GMR and TMR is that in TMR the current is a tunneling current and there is no conductance associated with the insulating barrier. From the point of view of the scattering theory this means that not only the match between the asymptotic wave-functions through the scattering region is important, but also how these wave-functions decay within the tunneling barrier.

III-4-3- Spin-transport through carbon nanotubes and nanowires

There was a considerable effort in combining the field of molecular and spin electronics. The typical device consists in a spin-valve that uses a molecular object as spacer. Potentially this has several advantages compared to more conventional materials since molecules are free of strong spin-flip scattering mechanisms such as spin-orbit and hyperfine interaction or scattering to magnetic impurities. In this race to organic spin-devices the use of carbon nanotubes occupies an important place.

Carbon nanotubes are almost defect-free graphene sheets rolled up in forming 1D molecules with enormous aspect ratios. Their conducting state depends on their chirality, however in the metallic configuration they are ideal conductors with a remarkably long phase coherence length. An important aspect is that the relevant physics at the Fermi level is entirely dominated by the p_x orbitals, which are radially aligned with respect to the tube axis. These include the bonding properties with other materials and between tubes. Therefore carbon nanotubes appear as an ideal playground for investigating both GMR and TMR through molecules. In fact one can expect that two tubes with different chirality will bond to a magnetic surface in a similar way, allowing us to isolate the effects of the molecule from that of the contacts. Indeed TMR-like transport through carbon nanotubes has been experimentally reported by several groups. [18,19,20,21,22]

The transport through an interface between such a magnetic metal and the nanotube is determined by the overlap between the corresponding Fermi surfaces. Three possible scenarios are possible. First the Fermi-wave vector of the carbon nanotube is smaller than both $k_F^{\uparrow}$ and $k_F^{\downarrow}$ ***(Figure III-5-a)***. In this case in the magnetic metal there is always a k-vector that matches the Fermi-wave vector of the nanotube for both spins. Therefore both spins can be injected into the tube and the total resistance will be small and spin-independent. Secondly the Fermi-wave vector of the carbon nanotube is larger than both $k_F^{\uparrow}$ and $k_F^{\downarrow}$ ***(Figure III-5-b)***. In this case there are no available states in the metallic contact whose wave-vectors match exactly the Fermi wave-vector of the carbon nanotube. Therefore in the zero-bias zero temperature limit the resistance is infinite. Nevertheless as one increases the temperature, phonon assisted transport starts to be possible. Spin electrons can be scattered out of the Fermi surface into states with large longitudinal momentum. At temperature T the fraction of electrons with energy above E_F is simply proportional to the Fermi distribution function. However, because of the exchange energy, spin-up electrons will possess higher momentum than spin-down. Therefore one can find more spin-up states with a longitudinal momentum matching the one of the nanotube than spin-down states. This gives a temperature-induced spin-dependent resistance. Hence one should expect that the increase of the temperature will decrease the resistance for spin-up electrons, leaving unchanged that of spin-down electrons.

Finally if the Fermi wave-vector of the carbon nanotube is larger than $k_F^{\downarrow}$ but smaller than $k_F^{\uparrow}$ ***(Figure III-5-c)***, only the majority electrons can enter the nanotube and the system becomes fully spin-polarized. In this situation a spin-valve structure made by magnetic contacts and carbon nanotube as spacer is predicted to show an infinite GMR at zero temperature, similar to the case of the half-metals. The increase of the temperature will produce a degradation of the polarization because also the spin-down electrons may occupy high energy states with large longitudinal momentum. Both the spins can be injected and the spin-polarization will depend on the number of occupied states with longitudinal momentum matching the one of the nanotube.

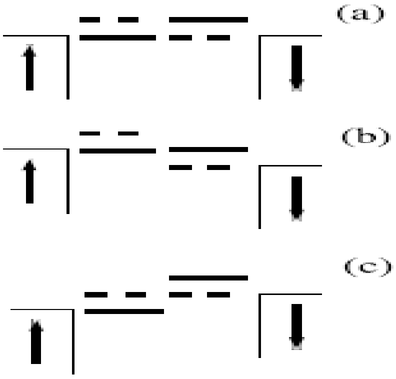


Figure III-5: Cartoon showing the levels alignment in the magnetic point contact. The solid (dashed) line denotes a majority (minority) spin molecular state. ***a)*** symmetric case at zero bias, ***b)*** symmetric case at positive bias, and ***c)*** symmetric case at negative bias. [18]

Two important aspects must be pointed out. First all these considerations are based on the assumption of perfectly crystalline systems. This may not be true in reality and the effects of breaking the translational invariance must be considered. From a qualitative point of view disorder will smear the Fermi surface and eventually produce some states with large longitudinal momentum. This will improve the conductance through the nanotube, even if its spin-polarization will be in general dependent on the nature of disorder.

Second, in contacts made from transition metals the simple parabolic band model introduced here is largely non-realistic. The Fermi surface of magnetic metal can comprise different manifolds with different orbital components and the degree of polarization of a junction depends upon how the different manifolds couple to the nanotube. In this case, simple theories are only speculative and more realistic bandstructure calculations are needed. These are rather problematic since the problem includes the need of describing transition metal leads and a molecule comprising a large number of degrees of freedom.

III-5- Conception of quantum computers and DiVincenzo criteria:

The fields of semiconductor physics and electronics have been successfully combined for many years. The invention of the transistor meant a revolution for electronics and has led to significant development of semiconductor physics and its industry. More recently, the use of the spin degree of freedom of electrons, as well as the charge, has attracted great interest. In addition to applications for spin electronics (spintronics) in conventional devices, for instance based on the giant magneto-resistance effect and spin-polarized field-effect transistors, there are applications that exploit the quantum coherence of the spin. This was encouraged by ground breaking experiments that showed coherent spin transport over long distances in semiconductors and long electron-spin dephasing times, on the order of 100 nanoseconds. In addition, spin-polarized carrier injection from magnetic to non-magnetic semiconductors has been demonstrated. Since the electron spin is a two-level system, it is a natural candidate for the realization of a quantum bit (qubit). The confinement of electrons in semiconductor structures like quantum dots allows for better control and isolation of the electron spin from its environment. Control and isolation are important issues to consider for the design of a quantum computer.

The successful implementation of a quantum computer demands that some basic requirements be fulfilled. These are known as the DiVincenzo criteria [35] and can be summarized in the following:

1- Information storage–the qubit: We need to find some quantum property of a scalable physical system in which to encode our bit of information, that lives long enough to enable us to perform computations.

2- Initial state preparation: It should be possible to set the state of the qubits to 0 before each new computation.

3- Isolation: The quantum nature of the qubits should be tenable; this will require enough isolation of the qubit from the environment to reduce the effects of decoherence.

4- Gate implementation: We need to be able to manipulate the states of individual qubits with reasonable precision, as well as to induce interactions between them in a controlled way, so that the implementation of gates is possible. Also, the gate operation time τ_G has to be much shorter than the decoherence time τ_D, so that $\tau_G/\tau_D \ll$r, where r is the maximum tolerable error for quantum error correction schemes to be effective.

5- Readout: It must be possible to measure the final state of our qubits once the computation is finished, to obtain the output of the computation.

To construct quantum computers of practical use, we emphasize that the scalability of the device should not be overlooked. This means it should be possible to enlarge the device to contain many qubits, while still adhering to all requirements described above. In this respect, very promising schemes for quantum computation are the proposals based on solid-state qubits, which could take advantage of existing technology.

III-6- Experimental achievements

III-6-1- Single and coupled quantum dots

We first discuss different experimental approaches to construct semiconductor quantum dot structures that enable control over the spin degree of freedom on the level of a single electron. The precise control of the number of excess electrons in a quantum dot is a necessary prerequisite to achieve control over the spin states of interest. The addition of an electron from the surrounding material to a negatively charged dot requires the charging energy δe_c to overcome the electrostatic energy of other electrons in the dot. The charging energy δe_c depends on the number N of charges confined in the dot. The regime (gate voltages) where the injection of additional electrons into the dot is blocked due to δe_c is known as the Coulomb blockade regime (***Figure III-6***). In recent years, a great deal of experimental effort has focused on the single-electron regime (N = 1) using different types of quantum dot structures. This regime provides experimental access to a spin 1/2 in the dot.

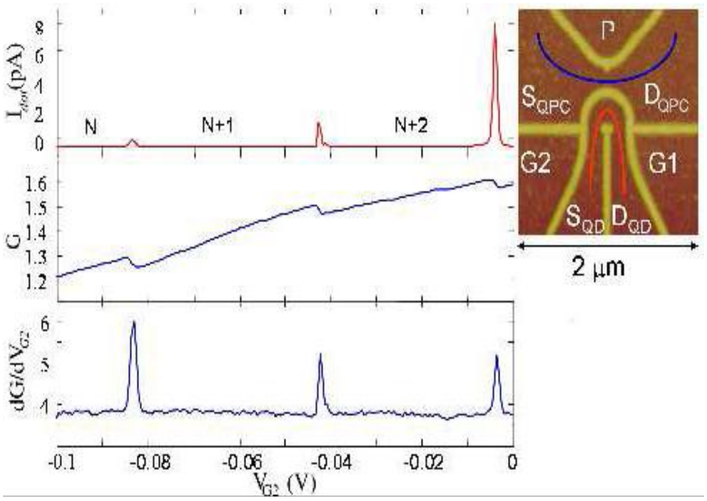


Figure III-6: Device (right) used to read-out the charge state of a quantum dot with a quantum point contact (QPC) [25]

As an experimental achievement of this, quantum dots can be created by electrical gating of a 2DEG (two dimensional electron gas) via lithographically defined gate electrodes (***Figures III-7, III-8***). Applying a negative voltage to the gates depletes the 2DEG underneath them, such that quantum dots are formed in the regions surrounded by the gates. Electrically gated dots are typically characterized by an electron level spacing δe ≈ 0.1 . . . 2 meV, a charging energy δe_c ≈ 1 . . . 2 meV, and a dot diameter l ≈ 10 . . .1000 nm. Typical materials for such dots include GaAs, InSb, and Si. Control of the coupling of electrically gated GaAs quantum dots has been demonstrated and investigated in-depth in transport experiments.

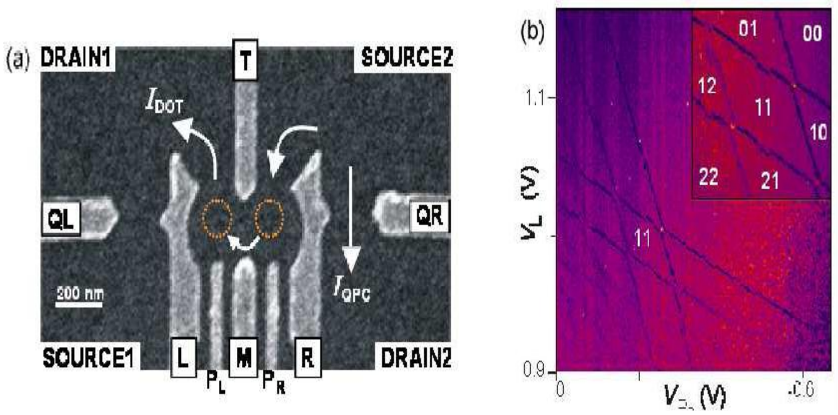


Figure III-7: ***(a)*** Scanning electron micrograph of a gated double dot structure with two adjacent quantum point contacts (QPCs). ***(b)*** Charge stability (“honeycomb”) diagram of the double quantum dot. [26]

As an alternative to electrical gating, etching techniques can also be applied to achieve lateral confinement in the plane of a 2DEG. For example, Tarucha et al. have produced gated vertical quantum dots by etching a pillar structure which contained a double-barrier heterostructure with an InGaAs quantum well as the 2DEG. ***Figures III-8***, ***III-9*** show structures containing dots of this type.

Further, quantum dot structures can be grown by self-assembly, e.g., using the Stranski-Krastanov growth technique [26]. In this technique, self-assembled dot islands form spontaneously during epitaxial growth due to a lattice mismatch between the dot and the substrate material. Typical sets of dot/substrate materials are InAs/GaAs, Ge/Si(100), GaN/AlN, InP/GaInP, and CdSe/ZnSe. The electron level spacing of this type of dot is typically δe ≈ 30 . . . 50 meV with a charging energy δe_c ≈ 20 meV, a diameter l ≈ 10 . . . 50 nm, and a height d ≈ 2 . . . 10 nm of the dot. Small selfassembled dots typically have a pyramidal shape with four facets, whereas larger dots (containing, e.g., 7 monolayers of InAs) form multi-faceted domes.

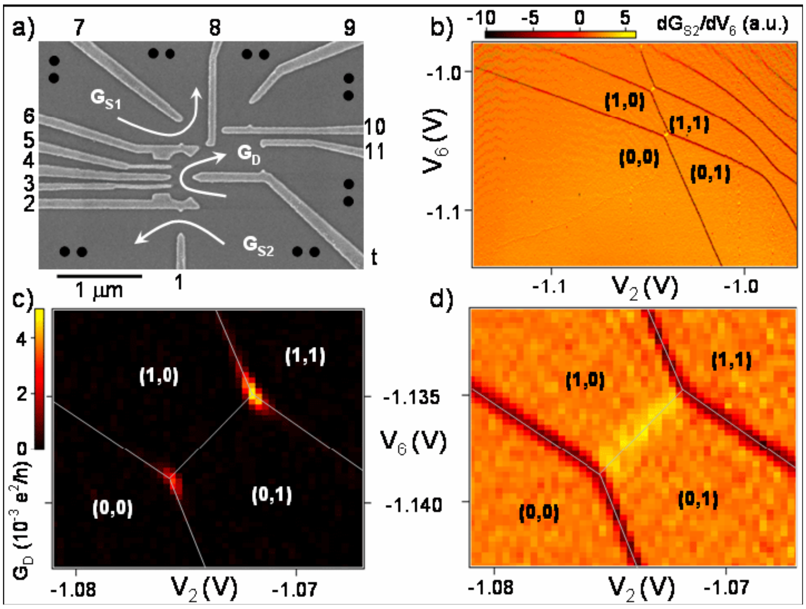


Figure III-8: (a) SEM micrograph of an electrically gated double quantum dot structure with neighbouring QPC charge detectors. ***(b)*** Large-scale plot of the differential conductance dGS2/dV6 as a function of the voltages V2 and V6 applied to gates 2 and 6, respectively. In ***(c)*** and ***(d)***, GD and dGS2/dV6 are shown, respectively, as a function of V2 and V6 in the region close to the (1,0) to (0,1) transition. [27]

If pyramidal self-assembled dots are covered with a thin layer of the substrate material (called the capping layer), the capped dots take-on an elliptical (or rarely, even a circular) shape. Additionally, these dots exert strain on the capping layer. If quantum dots are grown on the capping layer, they tend to grow on the strain field on top of the capped dots rather than at random positions. This enables the growth of vertically coupled quantum dots, where the thin capping layer acts as a barrier between the two dots (***Figure III-10-a***). A typical difficulty related to Stranski-Krastanov self assembled dots is the intrinsic randomness of the growth process, as shown in ***Figure III-10-b***. Yet, prepatterning of the substrate has been shown to be a way to achieve a well-defined growth position of the first dot layer (***Figure III-10-c***), paving the way to site-controlled arrays of single or coupled dots.

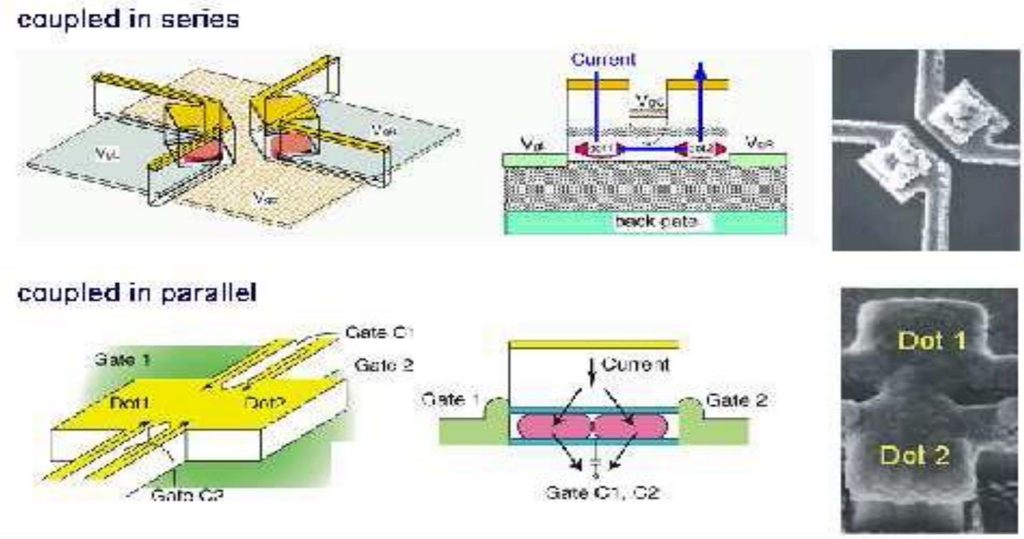


Figure III-9: Different designs for etched structures of coupled quantum dots. [28]

III-6-2- Charge and spin control in quantum dots

Precise control over the number of confined electrons has been demonstrated several years ago in InGaAs self-assembled dots, in gated vertical quantum dots, and also in electrostatically defined single and double dots in GaAs. The single-electron states of quantum dots in the low-energy range have been shown to be in agreement with a shell model. Because the quantum dot confinement is much stronger along the growth direction than perpendicular to it, the dot potential is effectively two-dimensional. The low-lying confined electron states can be well-approximated by the states of a two-dimensional harmonic oscillator. Thus, the single-particle ground state has (s) symmetry and the first excited shell has (p) symmetry. If an external magnetic field is applied perpendicular to the quantum dot plane, new harmonic oscillator states (Fock-Darwin states) are the exact eigenfunctions, with a frequency that increases with the magnetic field [30].

The degeneracy of the two spin states $\left|\uparrow\right\rangle$ and $\left|\downarrow\right\rangle$is lifted in the presence of a magnetic field due to the Zeeman interaction. This makes the two states energetically distinguishable (***Figure III-11***). The precise control of the occupation number of electrons in single and double quantum dots has enabled experiments on single spins in quantum dots.

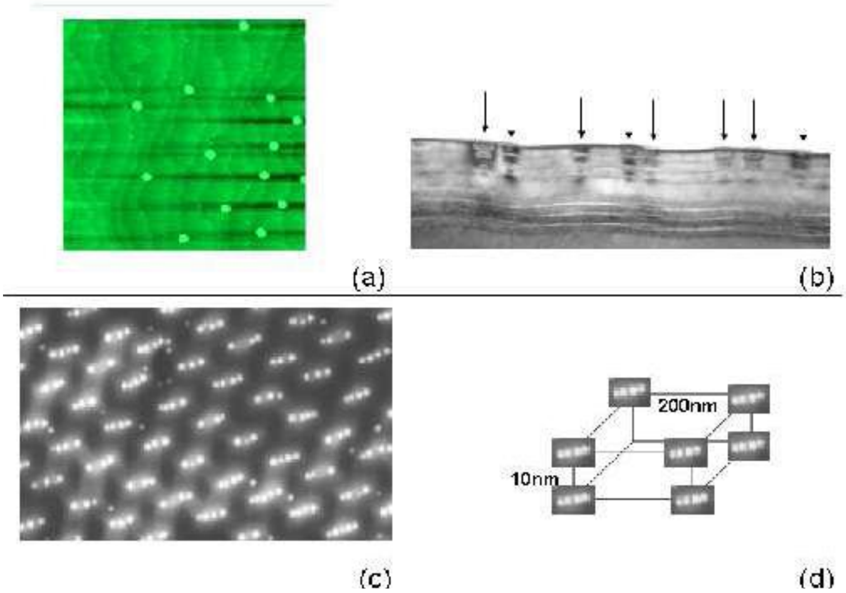


Figure III-10: Self-assembled InAs quantum dot structures. ***(a)*** AFM picture of dots grown at random locations. ***(b)*** Transmission electron microscope (TEM) cross-section of vertically stacked dots (indicated by arrows), ordered along the growth axis. ***(c)*** AFM picture of laterally ordered dots. This image was generated after prepatterning of the substrate. ***(d)*** Sketch of a three-dimensional lattice of dots that could be obtained by combining the growth methods of ***(b)*** and ***(c)***. [29]

III-6-3- Spin relaxation

Recently, expectations for the stability of spin qubits in quantum dots have grown considerably as progressively longer spin lifetimes have been reported. A series of works on electron spin relaxation in quantum dots started with Fujisawa et al. who reported a triplet-to-singlet relaxation time of τ_{S-T} = 200 μs in vertical quantum dots. More recently, a lower bound on the singlet-triplet relaxation time has been measured in lateral dots, giving $\tau_{S-T} \geq 70$ μs. Very quickly thereafter, a substantially longer relaxation time (τ_{S-T} = (2.58 ± 0.09) ms) was measured independently using a novel spin readout technique. Several groups have since measured T_1 for single electron spins. For electrostatically-defined GaAs dots, Hanson et al. [27] have reported a lower bound $T_1 \geq 50$ μs at a magnetic field of B = 7.5 T which was subsequently topped by Elzerman et al [29], with $T_1 \approx (0.85 \pm 0.11)$ ms at B = 8 T. In these experiments, a two-level pulse technique for the quantum dot gate voltage has been applied to inject an electron into the dot and to extract it later. In a certain parameter range, the Zeeman splitting of the two spin states is sufficient that tunnelling into or out of the dot is not possible for one of the two spin states (***Figure III-11***). This enables spin detection via the detection of charge in the quantum dot, which has been realized through an adjacent quantum point contact (QPC). In these experiments, the QPC (Quantum Point Contact) has been tuned via a gate voltage to a conductance G ≈ e^2/h, where the modulation of the current IQPC through the QPC has maximum sensitivity to changes in the electrostatic environment, including the number of charges in the quantum dot. Recently, Kroutvar et al. [29] established a lower bound $T_1 \geq 20$ ms at T = 1 K and B = 4 T for In(Ga)As self-assembled dots. In this experiment, an optical charge storage device has been excited with circularly polarized laser excitation. The larger level spacing of self-assembled dots (compared to gated GaAs dots) is responsible for the longer T1-time seen in this experiment which is limited by spin-orbit coupling.

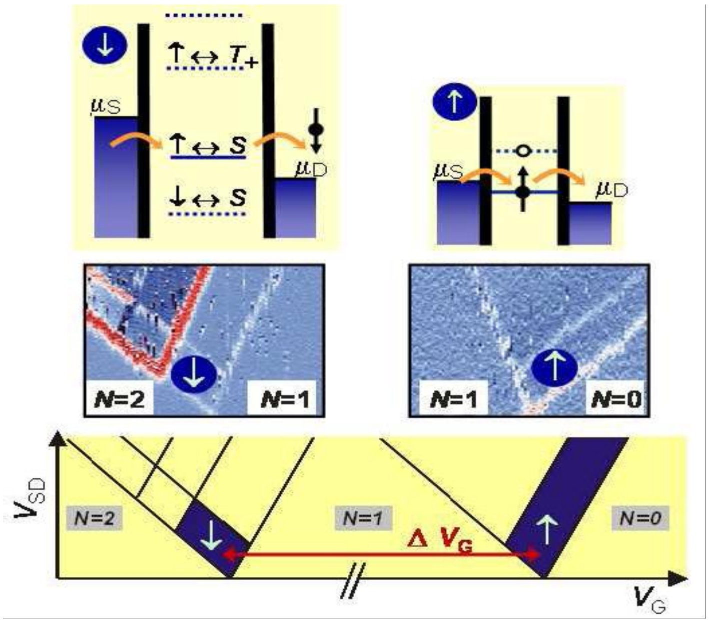


Figure III-11 : Quantum dot spin filter. ***(a)*** Only electrons in the state $\left|\downarrow\right\rangle$ are transported through the dot. ***(b)*** Only the spin ground state $\left|\uparrow\right\rangle$, can pass through the (empty) dot. In ***(c)*** and ***(d)***, the measured differential conductance dI/dV_{SD} is shown for the cases ***(a)*** and ***(b)***, respectively, with tunnelling current I and source-drain voltage V_{SD}. In ***(e)***, we show a scheme of the theoretically predicted dI/dV_{SD} [30]

III-6-4- Optical interaction and optical readout of spins

In this section, we first sketch some basics of optical transitions in quantum dots and then focus on the optical detection of spin states. The currently very active field of ultrafast laser technology suggests that single spin states can be optically detected and manipulated within very short times (picoseconds or even femtoseconds), several orders of magnitude faster than in schemes based on the transport of electric charge. Via the absorption of a photon, an electron in a confined valence-band state can be excited to a confined conduction-band state. For such inter-band transitions, optical selection rules apply and establish conditions on the quantum numbers of the optically coupled states. Provided the spin-orbit interaction is nearly isotropic, then it is a good approximation that the total angular momentum squared, $J^2 = (L + S)^2$, provides a good quantum number in semiconductors. Photons with circular polarization $\sigma^{\pm}$ carry an angular momentum with projection ±1 (in units of $\hbar$) along their propagation direction. For optical interactions, the total angular momentum is conserved, linking the spin of electrons and the polarization of photons. For a two-dimensional quantum dot with circular confinement, the z component J_z of J is a good quantum number. When J_z is a good quantum number in GaAs or InAs dots, the energetically lowest optical excitation at zero magnetic field typically includes two degenerate valence band states with total angular momentum projections $J_z = \pm 3/2$, which are also called heavy-hole (hh) states. A circularly polarized photon that is irradiated along the quantization axis z of J can excite one of the hh states to one of the conduction-band states with spin +1/2 or −1/2. For a given circular polarization, only one combination of these states satisfies the selection rules. This leads to a direct correspondence between the circular polarization of the photon and the spin of the optically excited electron. Taking advantage of this for the readout of spin states, light-emitting diodes ("spin-LEDs") have been fabricated, where the polarization of the emitted photons indicates the spin polarization of the electrons (or holes) injected into the spin-LED. A further step in nanoscale photonic and electronic technology has been taken recently by the growth of semiconductor nanowire superlattices. By modulating the reactants during catalytic growth of a nanowire, the nanowire finally consists of segments of different materials, e.g., Si and SiGe, InAs and InP, or GaAs and GaP. By alternating the two different materials, a superlattice can be formed. The combination of n- and p-type semiconductors, e.g., n-Si and p-Si or n-InP and p-InP, enables the bottom-up assembly of nanoscale (spin-)LEDs.

III-6-5- Spin initialization

To initialize the spin qubits, a strong polarization can be achieved by applying a strong magnetic field B, such that the Zeeman splitting is larger than the thermal energy. Further, electrons with parallel spins can also be injected via spin-polarized currents. The injections of spins from ferromagnetic semiconductors into normal semiconductors have been reported with polarizations up to 90%. Initialization can also be achieved using a spin filter or by optical schemes.

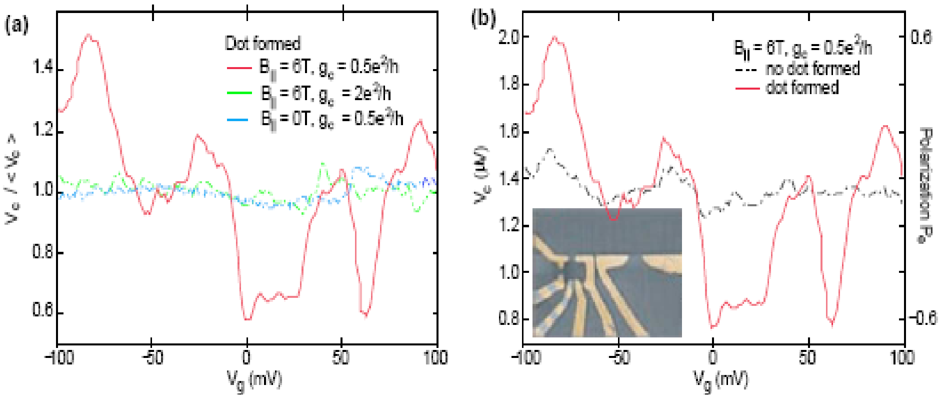


Figure III-12: Experimental demonstration of a spin filter. The figures (a) and (b) show the focusing peak height as a function of the quantum dot gate voltage Vg. [30]

III-7- Proposals for quantum computing

The first proposals for quantum computing made use of cavity quantum electrodynamics (QED), trapped ions, and nuclear magnetic resonance (NMR). All of these proposals benefit from potentially long decoherence times, relative to their respective gating times. The long decoherence times for these proposals and existing experimental expertise led to quick success in achieving experimental realizations. A conditional phase gate was demonstrated early-on in cavity-QED systems. The two qubit controlled “Not gate”, which, along with single-qubit rotations allows for universal quantum computation has been realized in single-ion and two-ion versions. The most remarkable realization of the power of quantum computing to date is the implementation of Shor's algorithm to factor the number 15 in a liquid-state NMR quantum computer. In spite of their great successes, the proposals based on cavity-QED, trapped ions and NMR may not satisfy the first DiVincenzo criterion (see chapter III-5). Specifically, these proposals may not meet the requirement that the quantum computer can be scaled-up to contain a large number of qubits. The requirement for scalability motivated the Loss-DiVincenzo proposal for a solid-state quantum computer based on electron spin qubits. This proposal was quickly followed by a series of proposals for alternate solid-state realizations and realizations for trapped atoms in optical lattices that may also be scalable.

III-7-1- Quantum dot quantum computing

The qubits of the Loss-DiVincenzo quantum computer are formed from the two spin states $(|\uparrow\rangle, \langle\downarrow|)$ of a confined electron. The original proposal focuses on electrons localized in quantum dots. These dots are typically generated from a two-dimensional electron gas, in which the electrons are strongly confined in the vertical direction. Lateral confinement is provided by electrostatic top gates, which push the electrons into small localized regions (***Figures III-13-a and III-14-b***).

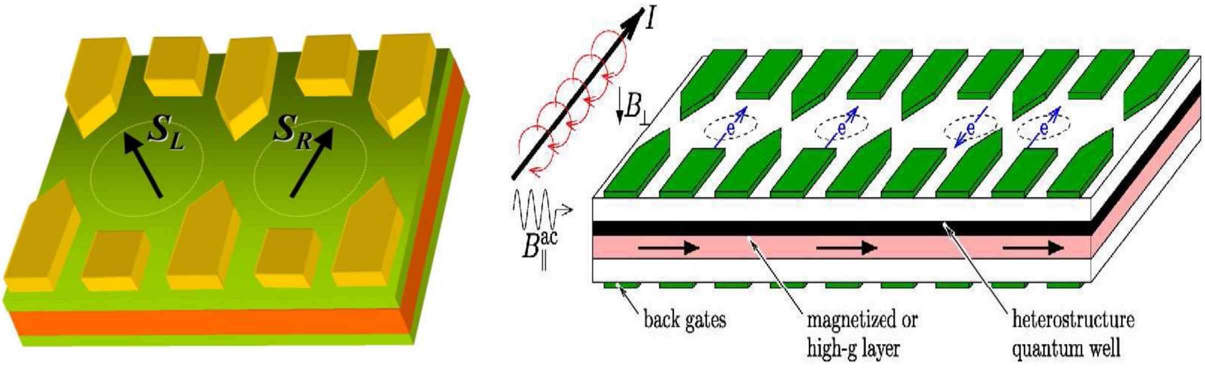


Figure III-13: **a-** Two neighbouring electron spins confined to quantum dots **b-** An array of exchange-coupled quantum dots. [31]

Initialization of the quantum computer can be achieved by allowing all spins to reach their thermodynamic ground state at low temperature T in an applied magnetic field B i.e., virtually all spins will be aligned if the condition

$$|g\mu_B B| \succ\succ k_B T \quad \ldots \tag{III-22}$$

is satisfied g factor , μ_B is the Bohr magneton, and k_B is the Boltzmann's constant. Single-qubit operations can be performed, in principle, by changing the local effective Zeeman interaction at each dot individually. To do this may require large magnetic field gradients, g-factor engineering, magnetic layers (***Figure III-13-b***), the inclusion of nearby ferromagnetic dots, polarized nuclear spins, or optical schemes. In the Loss-DiVincenzo proposal, two-qubit operations are performed by pulsing the electrostatic barrier between neighbouring spins. When the barrier is high, the spins are decoupled. When the inter-dot barrier is pulsed low, an appreciable overlap develops between the two electron wave functions, resulting in a non-zero Heisenberg exchange coupling J. The Hamiltonian describing this time-dependent process is given by

$$H(t) = J(t)S_L \cdot S_R. \quad \ldots \qquad \textit{(III-23)}$$

wher S_R is the right spin and S_L is the left spin. This Hamiltonian induces a unitary evolution given by the operator

$$U = T \exp\left\{-i\int H(t)\, {dt}/{\hbar}\right\}, \quad \ldots \qquad \textit{(III-24)}$$

where T is the time-ordering operator. If the exchange is pulsed on for a time τ_G such that

$$\int J(t)\, {dt}/{\hbar} = J_0\, {\tau_s}/{\hbar} = \pi\,, \quad \ldots \qquad \textit{(III-25)}$$

the states of the two spins, with associated operators S_L and S_R, as shown in ***(Figure III-13-a)***, will be exchanged. This is the swap operation. Pulsing the exchange for the shorter time $\tau_G/2$ generates the "square-root of swap" operation, which can be used in conjunction with single qubit operations to generate the controlled-Not (see chapter IV) gate.

In addition to the time scale τ_G, which gives the time to perform a two-qubit operation, there is a time scale associated with the rise/fall-time of the exchange J(t). This is the switching time τ_{sw}. When the relevant two-spin Hamiltonian takes the form of an ideal (isotropic) exchange, the total spin is conserved while switching.

However, to avoid jumps to higher orbital states during gate operation, the exchange coupling must be switched adiabatically. More precisely,

$$\tau_{sw} \succ\succ 1/\omega_0 \approx 10^{-12}\ \mathrm{s}, \quad \ldots \qquad \textit{(III-26)}$$

where $\hbar\omega_0 \approx 1\text{meV}$ is the energy gap to the next orbital state. We stress that this time scale is valid only for the ideal case of a purely isotropic exchange interaction. When the exchange interaction is anisotropic, different spin states may mix and the relevant time scale for adiabatic switching may be significantly longer. For scalability, and application of quantum error correction procedures in any quantum computing proposal, it is important to turn off inter-qubit interactions in the idle state. In the Loss-DiVincenzo proposal, this is achieved with exponential accuracy since the overlap of neighbouring electron wave functions is exponentially suppressed with increasing separation.

III-7-2- Quantum computing and the quantum Hall effect

Based on observed long lifetimes for nuclear spin states, Privman et al. [31] have proposed a quantum computer composed of nuclear spins embedded in a two dimensional electron gas (2DEG) in the quantum-Hall regime. The qubits of their proposal are encoded in the states of nuclear spins, which must be sufficiently separated to avoid dipolar coupling, but close enough ($10\approx$nm) to allow significant interaction via the electron gas. Initialization of the qubits is achieved by placing spin-polarized conducting strips with a current of electrons above the nuclear spin qubits. The contact hyperfine interaction between electron and nuclear spins causes a polarization transfer from the electrons in the strips to the nuclear spins, preferentially orienting the nuclear spins along the electron spin polarization direction. Readout is performed in a complementary manner, with a transfer of polarization from the nuclear spins to electrons in the conducting strips. Single-qubit operations are performed via standard NMR pulses, which would require strong magnetic field gradients or many different nuclear spin species to bring single specific nuclear spins into resonance, while leaving the other qubits unchanged. A pairwise interaction between the nuclear spin qubits is necessary for the implementation of two-qubit gates. This interaction is generated by a superexchange, mediated by electrons in the quantum Hall fluid that surrounds the nuclear spins (***Figure III-14-c***). The electron gas that couples the nuclear spins should be in the quantum Hall regime to avoid Friedel oscillations in the electron density. To perform computations, it is necessary to switch the interaction on and off. In the original work of Privman et al, it was not clear how best to pulse the inter-qubit interaction. Topics such as switching error and perhaps the most important of all, decoherence, are not addressed in the original work of Privman et al. However, subsequent studies of the decoherence of nuclear spins in the integer quantum Hall regime have led to the prediction that the decoherence time for these qubits could be as long as $\tau_D \approx 10^{-1}$ s.

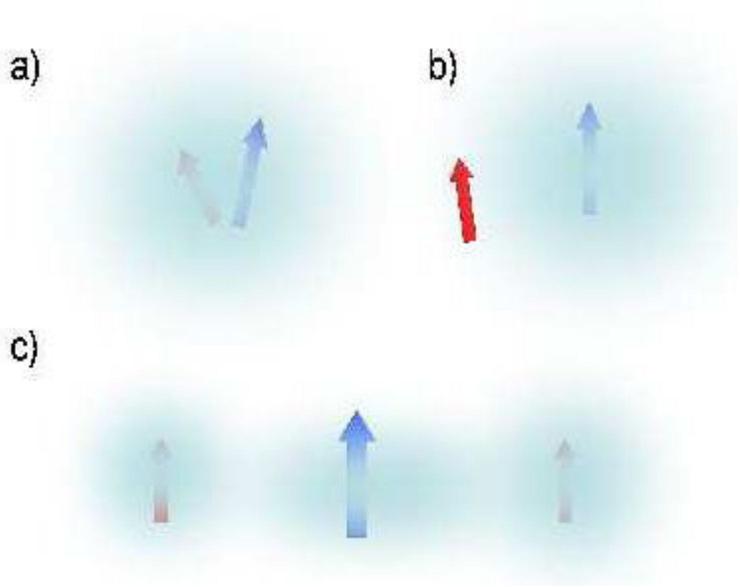


Figure III-14: Schematic diagram illustrating the Fermi contact hyperfine interaction. [31]

III-8- Obstacles to quantum dot quantum computing

Several major obstacles to quantum dot quantum computation were identified and addressed in the original work of Loss and DiVincenzo, and later elaborated upon. These obstacles include entanglement, gating error, and perhaps most importantly, coherence.

III-8-1- Flying qubits and entanglement generation

In addition to the five DiVincenzo criteria for quantum computation, there are two “desiderata”, which are important for performing quantum communication tasks. These desiderata are summarized in the following statements:

1- The ability to inter-convert stationary and flying qubits.

2- The ability to faithfully transmit flying qubits between distant locations.

The whimsical term “flying qubits” refers to qubits that can be conveniently moved from place to place. The most obvious choice for a flying qubit is provided by the polarization states of photons. In the context of quantum-dot quantum computing, this has led to a number of proposals for the conversion of quantum information or entanglement from spin to light, and vice versa. More recent work has suggested that “free electron quantum computation” may be possible in principle, in which mobile electrons travelling between dots could replace photons as the flying qubit medium of choice.

Deeply connected to the implementation of flying qubits is the creation of nonlocal entanglement. The race to create and measure entangled particle pairs has led to a virtual industry of so-called “entangler” proposals for the spin and orbital degrees of freedom. These proposals have the very ambitious goal of generating and spatially separating a many-particle quantum superposition that can not be factorized into single-particle states. The canonical example of such a state for the spin degree of freedom is the singlet formed from two spin 1/2 particles: $\left(\left|\uparrow\downarrow\right\rangle - \left|\downarrow\uparrow\right\rangle / \sqrt{2}\right)$. The various efforts related to spin entanglement include proposals to extract and separate spin-singlet pairs from a superconductor through two quantum dots or Luttinger-liquid leads and proposals that generate entanglement near a magnetic impurity, through a single dot, from biexcitons in double quantum dots, through a triple dot, and from Coulomb scattering in a two-dimensional electron gas. Entanglement generation and measurement remains a lofty goal for those working on solid-state quantum computing, theorists and experimentalists alike. Recent experiments that have measured the concurrence (an entanglement measure) for electrons in the ground state of a two-electron quantum dot point to a promising future for entanglement-related phenomena in the solid state.

III-8-2- Gating error

Hu and Das Sarma [48] have evaluated the probability for double-occupancy of one of the dots in the Loss-DiVincenzo proposal using Hartree-Fock and molecular orbital techniques. They suggest that it may be difficult to achieve both a significant exchange coupling and low double-occupancy probability. Schliemann et al.[31] and more recently Requist et al. [31] have investigated the probability for double-occupancy gating errors in a pair of coupled quantum dots during swap gate operation. Through numerical and analytical study they have found that the Loss-DiVincenzo proposal is very robust against double-occupancy errors when operated in the adiabatic regime. Barrett and Barnes [31] have subsequently shown that orbital dephasing can result in a significant error rate (10^{-2}–10^{-3} errors per gate operation). This is comparable to current estimates for the maximum error rate allowable for quantum error correction to be effective, but further studies on the nature of the spin-orbit interaction have suggested that the spin-orbit coupling can be minimized with careful pulsing of the exchange during gate operations. When the potential barrier between quantum dots is pulsed low, the overlap between nearest neighbour dots is appreciable, while that between next-nearest and next-next-nearest neighbours is exponentially suppressed with distance. In spite of the smallness of these interactions, Mizel and Lidar have recently suggested that three- and four-spin interactions in a realistic quantum computing proposal may lead to substantial gating errors. These problems are, however, specific to a particular architecture, and it is possible that they could be corrected or exploited by adjusting the device design.

III-9- Future Goals

III-9-1- Detection of single-electron spin decoherence

After the recent successful measurements of the T_1-lifetime of single electron spins in quantum dots, measurements of the decoherence time T_2 are due. To achieve such an experiment, an initial coherent evolution of the electron spin must be produced. This can be done, e.g., with electron spin resonance (ESR) or by inducing spin precession in a transverse magnetic field. The decay of the spin coherence can then be measured. Several proposals of this type have been made. Engel and Loss [30] have proposed a measurement of the sequential tunnelling current through a dot containing a single electron spin in the presence of ESR excitation. Sequential tunnelling, in general, describes a regime where charge transport only occurs via a sequence of first-order tunnelling processes. In the regime when sequential tunnelling is only possible via an intermediate singlet state on the dot, the stationary current I is a Lorentzian as a function of the ESR detuning $\delta_{ESR} = \omega_{ESR} - g_e\mu_B B$, where ω_{ESR} is the ESR frequency. The inverse of the linewidth of $I(\delta_{ESR})$ provides a lower bound for the intrinsic T_2 time of a single electron spin. Further, the coherent Rabi oscillations due to ESR pulses can be observed in the time-averaged current I(tp) as a function of the ESR pulse length tp. Gywat et al.[31] have theoretically studied the optical detection of magnetic resonance (ODMR) to measure the T_2-time of a single electron spin in a quantum dot. In this approach, the dot initially contains a single excess electron that is subject to ESR excitation. Unlike a tunnelling experiment, optical transitions are subject to selection rules and are not restricted to the Coulomb blockade regime, e.g., if the excess electron is present due to n-doping and is not electrically injected.

III-9-2- Single-qubit rotations

A further important step towards the goal of quantum computation is the implementation of a single-qubit gate. To achieve this for the Loss-DiVincenzo proposal, several possible strategies have been developed. The simplest way to rotate a spin is by applying a pulsed magnetic field. In an array of quantum dots, such fields could be applied to single spins. Further, in the presence of an rf magnetic field applied to an ensemble of electron spins, the tenability and precise control of the individual Zeeman splittings is sufficient to produce single spin rotations. When the ESR resonance condition is matched, the spin rotates with maximum amplitude, according to the well-known Rabi formula. Detuning of the Zeeman splitting of an individual spin from the ESR resonance slows its precession frequency and the spin stops rotating entirely when the detuning is larger than the ESR linewidth. Control of the Zeeman splitting at the single-spin level is therefore another way to perform single-spin rotations. This can be achieved in principle by controlling local magnetic fields or local Overhauser fields. For a structure designed to apply ESR excitation to a single quantum dot (***Figure III-15***). Another approach is the individual control of the electron g-factor instead of the local magnetic field. Salis et al.[32] have demonstrated electrically controlled modulation of the g-factor in an AlGaAs quantum well containing a gradient in the Al concentration. Here, the electron wave function was shifted between regions with different Al concentration via applied gate voltages, which resulted in the observation of a different electron g-factor.

Alternative proposals to produce single-spin rotations are related to all-optical Raman transitions and stimulated Raman adiabatic passage (STIRAP), a method based on two-photon Raman transitions which has already been applied to atoms and molecules to transfer a precisely controlled population between two quantum states. While Troiani et al.[32] have also considered the realization of conditional and unconditional quantum gates using an additional adjacent quantum dot, Chen et al.[32] have proposed a STIRAP process with no auxiliary state, but in the presence of a transverse magnetic field. In this setup, control of the relative phase and the relative intensity of two applied laser pulses enable an arbitrary spin rotation for a given polarization of the light and direction of the transverse magnetic field. As an alternative method of performing a spin rotation on an excess electron confined to a quantum dot, Calarco et al. [31] have proposed to excite lh states via a sequence of a linearly and then a circularly polarized laser π-pulse. Given this abundance of proposals for single-qubit gates, there is great hope for working experimental realizations in the near future.

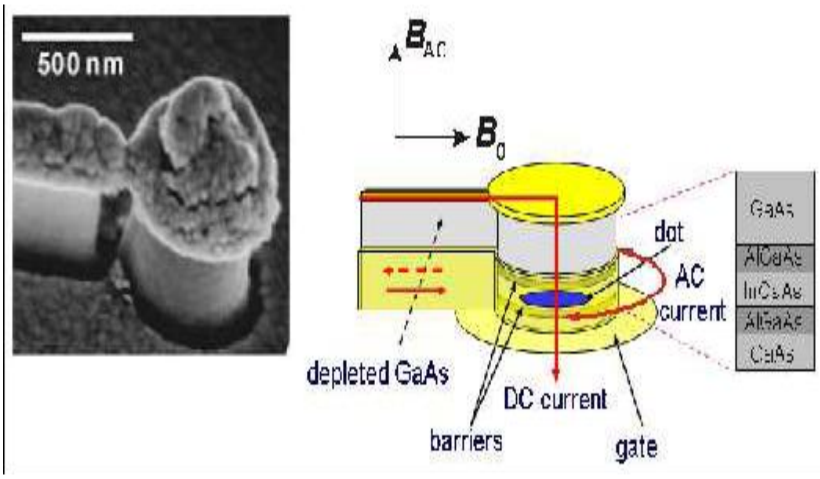


Figure III-15: SEM picture and scheme of a structure to apply a local rf magnetic field to a quantum dot. [32]

III-9-3- Two-Qubit Gates

Swapping of the spin states of two electrons located in closely spaced quantum dots seems by now to be a realistic first experimental step towards a two-qubit gate for spins. This can be achieved by controlling the overlap of the two wave functions of the electrons and thus the singlet-triplet splitting J. The interdot tunnel splitting and J can be determined from a transport experiment in the sequential tunnelling regime. Recently, J has been measured for two electrons in a single gated quantum dot by detecting inelastic cotunnelling above and below a magnetic field driven singlet-triplet transition. In the cotunnelling regime, only second-order tunnelling processes contribute to charge transport. Because the dot was elliptical, a two-electron wave function similar to that in a double dot was expected. Two different samples yielded J ≈ 0.2 meV and J ≈ 0.57 meV at B = 0. The critical magnetic field for the singlet-triplet transition (where J = 0) has been measured to be B^* ≈ 1.3 T. For the interaction parameter, φ ≈ 0.5 ± 0.1 has been obtained, indicating that the ground state given by $\left|+\uparrow,+\downarrow\right\rangle-\varphi\left|-\uparrow,-\downarrow\right\rangle \Big/ \sqrt{1+\varphi^2}$ (where ± stands for the symmetric/antisymmetric orbital wave function) consists of a singlet with a significant admixture of single-electron orbitals due to the electron-electron interaction. The entanglement of the two electron spins in the state above can be quantified by the concurrence $C = 2\varphi/(1 + \varphi^2)$. The experimental result C ≈ 0.8 shows that electron-electron interaction reduces the degree of spin entanglement from its maximum (C = 1), which is obtained for a singlet (having φ = 1). This demonstration strongly encourages that similar results might be soon obtained in double dots (which are needed for spatially separating the two qubits).

III-10- Conclusions

A decade after the discovery of the GMR effect the future of spin-electronics in nanoscale systems looks bright. This is mainly due to the improved understanding of the spin-transport mechanisms and the better control over the device processing. At the same time the possibility of conducting spin-transport measurements in systems comprising a handful of atoms has opened completely new prospectives. We can envision in a near future new devices where the spin and molecular functionalities will be combined achieving a broad range of applications, from biological sensors to tools for coherent quantum data processing.

From the theoretical side the last decade has also witnessed a rapid evolution of computational methods for both electronic structure and quantum transport. Several numerical implementations are currently available. The most advanced of them are based on ab initio schemes and therefore do not depend on parameters obtained from experiments. These open the way to a physics “without compromises”, where the numerical predictions must reproduce the experimental data, if the systems under investigation are the same. For this reason ab initio transport schemes have became invaluable tools. Certainly the future of modeling spin transport at the nanoscale is dawning.

The demonstration of working single and two-qubit gates and finally the production of quantum dot arrays that enable the application of an entire quantum algorithm including error correction are the major problems to tackle towards the goal of a solid-state implementation of quantum information processing.

IV-1- Introduction

Let us recall that Grover's algorithm is a quantum algorithm for searching an unsorted database with N entries in O($N^{1/2}$) time and using O(logN) storage space. It was invented by Lov Grover in 1996 [39].

Classically, searching an unsorted database requires a linear search, which is O(N) in time. Grover's algorithm, which takes O($N^{1/2}$) time, is the fastest possible quantum algorithm for searching an unsorted database. It provides "only" a quadratic speedup, unlike other quantum algorithms, which can provide exponential speedup over their classical counterparts. However, even quadratic speedup is considerable when N is large.

Like all quantum computer algorithms, Grover's algorithm is probabilistic, in the sense that it gives the correct answer with high probability. The probability of failure can be decreased by repeating the algorithm.

IV-2-1- The way to the quantum computation

Let us now have a closer look at the way a quantum computer works. We will do so by comparing the concepts of classical computing with the basics of quantum computing. In fact, many classical concepts have very similar quantum counterparts, like bits become qubits and still the logic is often best explained within a circuit model.

IV-2-2- Qubits and quantum parallelism

The elementary information carriers in a quantum computer are the qubits – quantum bits. In contrast to classical bits which take on either the value zero or one, qubits can be in every superposition of the state vectors $|0\rangle$ and $|1\rangle$. This means that the vector $|\psi\rangle$ describing the (pure) state of the qubit can be any linear combination $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ of the vectors $|0\rangle$ and $|1\rangle$ with complex coefficients α and β. In the same way a system of many qubits can be in a superposition of all classically possible states

$$|0,0,\ldots,0\rangle + |1,0,\ldots,0\rangle + \ldots + |1,1,\ldots,1\rangle \quad \ldots \qquad \textbf{\textit{(IV-1)}}$$

The basis $|0,0,\ldots,0\rangle, |1,0,\ldots,0\rangle, \ldots, |1,1,\ldots,1\rangle$ that corresponds to the binary words of length n in a quantum system of n qubits is called the computational basis. Using the superposition of Equation ***(IV-1)*** as an input for an algorithm means somehow to run the computation on all classically possible input states at the same time. This possibility is called quantum parallelism and it is certainly one of the reasons for the computational power of a quantum computer. The mathematical structure behind the composition of quantum systems is the one of the tensor product. Hence, vectors like $|0,0,\ldots,0\rangle$ should be understood as $|0\rangle \otimes \ldots \otimes |0\rangle = |0\rangle^{\otimes n}$. This implies that the dimension of the space characterizing the system grows exponentially with the number of qubits.

A Physically, qubits correspond to effective two-level systems like the ground state and excited state of an atom, the polarization degree of freedom of up-and down orientation of a spin 1/2 particle. Such a physical system can be in any pure state that can be represented by a normalized vector of the above form. A pure state of a composite quantum system that is not a product with respect to all constituents is called an entangled pure state.

IV-2-3- Readout and probabilistic nature of quantum computers

An important difference between classical and quantum computers lies in the readout process. In the classical case there is not much to say: the output is a bit-string which is obtained in a deterministic manner, i.e., repeating the computation will lead to the same output again. However, due to the probabilistic nature of quantum mechanics, this is different for a quantum computer. If the output of the computation is for instance the state vector $|\psi\rangle$, α and β cannot be determined by a single measurement on a single specimen. In fact, $|\alpha|^2$ and $|\beta|^2$ are the probabilities for the system to be found in $|0\rangle$ and $|1\rangle$ respectively. Hence, the absolute values of these coefficients can be determined by repeating the computation, measuring in the basis $|0\rangle$, $|1\rangle$ and then counting the relative frequencies. The actual outcome of every single measurement is thereby completely undetermined. In the same manner, the state of a quantum system consisting of n qubits can be measured in the computational basis, which means that the outcome corresponding to some binary word occurs with the probability given by the square of the absolute value of the respective coefficient. So in effect, the probabilistic nature of the readout process on the one hand and the possibility of exploiting quantum parallelism on the other hand are competing aspects when it comes to comparing the computational power of quantum and classical computers.

IV-3- The Quantum Circuit Model

In the quantum circuit model, we have logical qubits carried along ‘wires’, and quantum gates that act on the qubits. A quantum gate acting on *n* qubits has the input qubits carried to it by *n* wires, and *n* other wires carry the output qubits away from the gate. A quantum circuit is often illustrated schematically by a *circuit diagram* as shown in ***Figure IV-1***. The wires are shown as horizontal lines, and we imagine the qubits propagating along the wires from left to right in time. The gates are shown as rectangular blocks. For convenience, we will restrict attention to unitary quantum gates (which are also reversible). Recall that non-unitary (non-reversible) quantum operations can be simulated by unitary (reversible) quantum gates if we allow the possibility of adding an ancilla and of discarding some output qubits. A circuit diagram describing a superoperator being implemented using a unitary operator is illustrated in ***Figure IV-2.***

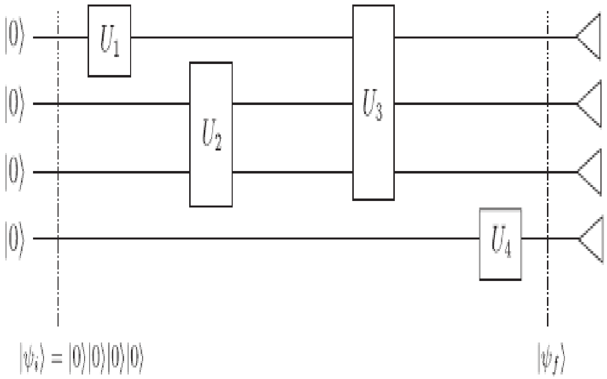


Figure IV-1: A quantum circuit. [33]

In the example of ***Figure IV-1***, the 4-qubit state $|\psi_i\rangle = |0\rangle \otimes |0\rangle \otimes |0\rangle \otimes |0\rangle$ enters the circuit at the left. These qubits are processed by the gates U_1, U_2, U_3, and U_4. At the output of the circuit we have the collective (possibly entangled) 4-qubit state$|\psi_f\rangle$. A measurement is then made of the resulting state. The measurement will often be a simple qubit-by-qubit measurement in the computational basis, but in some cases may be a more general measurement of the joint state. A measurement of a single qubit in the computational basis is denoted on a circuit diagram by a small triangle, as shown in ***FigureIV-1***.

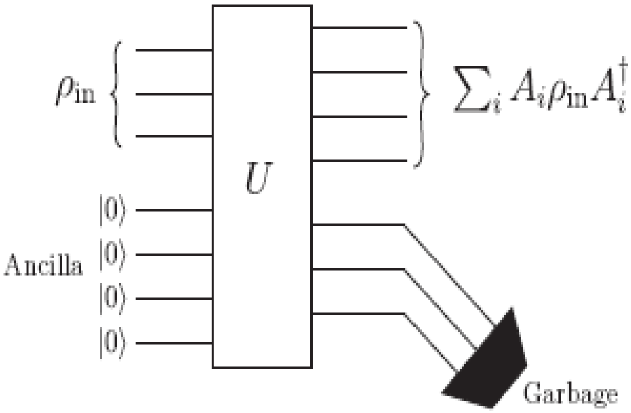


Figure IV-2: A general (possibly irreversible) quantum operation or superoperator can be realized using a unitary operation by adding an ancilla and tracing out part of the output. [33]

The triangle symbol will be modified for cases in which there is a need to indicate different types of measurements. Recall that the measurement postulate stated that a measurement outputs a classical label '*i*' indicating the outcome of the measurement and a quantum state$|\phi_i\rangle$. Thus, we could in general draw our measurement symbol with a 'quantum' wire carrying the quantum state resulting from the measurement, together with a classical wire carrying the classical label, as depicted in ***Figure IV-3***.

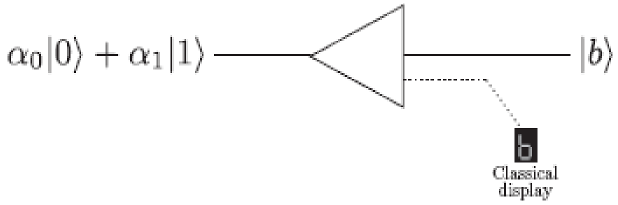


Figure IV-3: The measurement of the quantum state $\alpha_0|0\rangle + \alpha_1|1\rangle$results in a quantum output $|b\rangle$with probability $|\alpha_b|^2$ ($b \in \{0, 1\}$) together with a classical label 'b' indicating which outcome was obtained. [33]

Quite often, the quantum outcome is discarded or ignored, and we are only interested in the classical information telling us which outcome occurred. In such cases, we will not draw the quantum wire coming out of the measurement symbol. We will usually omit the classical wire from circuit diagrams as well.

IV-3-1- Quantum Gates

IV-3-1-1- Qubit Gates

We said in the second chapter that any unitary operator acting on a 2-dimensional quantum system (a qubit) is called a ‘1-qubit quantum gate’. We gave the “quantum NOT gate” (sometimes called the Pauli *X* gate) as an example. Every 1-qubit pure state is represented as a point on the surface of the Bloch sphere, or equivalently as a unit vector whose origin is fixed at the centre of the Bloch sphere. A 1-qubit quantum gate *U* transforms a quantum state$|\psi\rangle$ into another quantum state *U* $|\psi\rangle$. In terms of the Bloch sphere (see chapter II), the action of *U* on $|\psi\rangle$can be thought of as a rotation of the Bloch vector for $|\psi\rangle$ to the Bloch vector for *U* $|\psi\rangle$. For example, the not gate takes the state$|0\rangle$ to the state $|1\rangle$(and takes $|1\rangle$ to$|0\rangle$). In terms of the Bloch sphere, this action can be visualized as a rotation through an angle π about the *x* axis, as illustrated in ***FigureIV-4***.

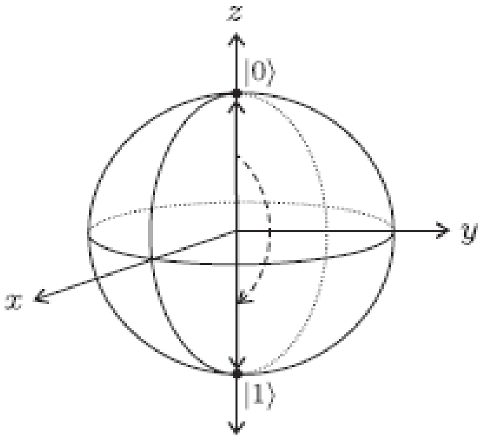


Figure IV-4: The NOT gate rotating the state $|0\rangle$ to the state$|1\rangle$. [34]

We saw in chapter II how to compute the exponential (and other functions) of operators. If we exponentiate the Pauli matrices, we get unitary operators corresponding to very important classes of 1-qubit gates. These are the *rotation gates*, which correspond to rotations about the *x*-,*y*-, and *z*- axes of the Bloch sphere. They are defined in terms of the Pauli gates, and so for convenience, we remind you now of the definitions of the Pauli gates:

$$I \equiv \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \qquad Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \qquad Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad \ldots \quad \textbf{\textit{(IV-2)}}$$

The rotation gates are defined as follows:

$$\begin{aligned} R_x &= e^{\frac{-i\theta X}{2}} \\ R_y &= e^{\frac{-i\theta Y}{2}} \qquad \ldots \qquad \textbf{\textit{(IV-3)}} \\ R_z &= e^{\frac{-i\theta Z}{2}} \end{aligned}$$

It is easy to check that the Pauli matrices *X, Y* , and *Z* satisfy the conditions $X^2 = I$, $Y^2 = I$, and $Z^2 = I$, and so we can write the rotation gates as:

$$\begin{aligned} R_x(\theta) = e^{\frac{-i\theta X}{2}} &= \cos\left(\frac{\theta}{2}\right) I - i\sin\left(\frac{\theta}{2}\right) X \\ R_y(\theta) = e^{\frac{-i\theta Y}{2}} &= \cos\left(\frac{\theta}{2}\right) I - i\sin\left(\frac{\theta}{2}\right) Y \qquad \ldots \qquad \textbf{\textit{(IV-4)}} \\ R_z(\theta) = e^{\frac{-i\theta Z}{2}} &= \cos\left(\frac{\theta}{2}\right) I - i\sin\left(\frac{\theta}{2}\right) Z \end{aligned}$$

Knowing the matrices for *I,X, Y* , and *Z* in the computational basis, we can now write the rotation gates as matrices in the computational basis:

$$R_x(\theta) = \begin{bmatrix} \cos\left(\frac{\theta}{2}\right) & -i\sin\left(\frac{\theta}{2}\right) \\ -i\sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{bmatrix}$$

$$R_y(\theta) = \begin{bmatrix} \cos\left(\frac{\theta}{2}\right) & -\sin\left(\frac{\theta}{2}\right) \\ \sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{bmatrix} \qquad \ldots \qquad \textbf{\textit{(IV-5)}}$$

$$R_z(\theta) = \begin{bmatrix} e^{-i\frac{\theta}{2}} & 0 \\ 0 & e^{i\frac{\theta}{2}} \end{bmatrix}$$

Consider an arbitrary 1-qubit state, written in terms of its Bloch vector angles σ and τ :

$$\cos\left(\frac{\sigma}{2}\right)|0\rangle + e^{i\tau}\sin\left(\frac{\sigma}{2}\right)|1\rangle \qquad \dots \qquad \textbf{(IV-6)}$$

In the computational basis, this can be written as the column vector

$$\begin{pmatrix} \cos\left(\frac{\sigma}{2}\right) \\ e^{i\tau}\sin\left(\frac{\sigma}{2}\right) \end{pmatrix} \qquad \dots \qquad \textbf{(IV-7)}$$

The effect of applying $R_z(\theta)$ on this state can be seen by performing a matrix multiplication:

$$\begin{bmatrix} e^{-i\frac{\theta}{2}} & 0 \\ 0 & e^{i\frac{\theta}{2}} \end{bmatrix} \begin{pmatrix} \cos\left(\frac{\sigma}{2}\right) \\ e^{i\tau}\sin\left(\frac{\sigma}{2}\right) \end{pmatrix} = \begin{pmatrix} e^{-i\frac{\theta}{2}}\cos\left(\frac{\sigma}{2}\right) \\ e^{i\frac{\theta}{2}}e^{i\tau}\sin\left(\frac{\sigma}{2}\right) \end{pmatrix}$$

$$= e^{-i\frac{\theta}{2}} \begin{pmatrix} \cos\left(\frac{\sigma}{2}\right) \\ e^{i\theta}e^{i\tau}\sin\left(\frac{\sigma}{2}\right) \end{pmatrix}$$

$$= e^{-i\frac{\theta}{2}}(\cos\left(\frac{\sigma}{2}\right)|0\rangle + e^{i(\tau+\theta)}\sin\left(\frac{\sigma}{2}\right)|1\rangle) \qquad \dots \qquad \textbf{(IV-8)}$$

Since a global phase is insignificant, we have the state

$$\cos\left(\frac{\sigma}{2}\right)|0\rangle + e^{i(\tau+\theta)}\sin\left(\frac{\sigma}{2}\right)|1\rangle \qquad \dots \qquad \textbf{(IV-9)}$$

We see that effect of $R_z(\theta)$ has been to change the angle τ to $\tau + \theta$, which is a rotation of θ about the z-axis of the Bloch sphere. To see that $R_x(\theta)$ and $R_y(\theta)$ implement rotations about the x- and y-axes of the Bloch sphere is trickier, because such rotations involve changes to both angles σ and τ .

It will be useful to show how to decompose any given 1-qubit gate into a sequence of rotations about the main axes of the Bloch sphere. The following theorem tells us that we can decompose any 1-qubit gate into a sequence of two rotations about the z-axis and one rotation about the y-axis, along with a suitable phase factor.

Theorem IV-1: [34]

Suppose U is a 1-qubit unitary gate. Then there exist real numbers α, β, γ, and δ such that

$$U = e^{i\alpha}R_z(\beta)R_y(\gamma)R_z(\delta) \quad \ldots \qquad \textbf{(IV-10)}$$

The proof of this follows from the fact that *U* is unitary, and the definition of the rotation matrices. There is nothing special about the *y*- and *z*-axes of the Bloch sphere. We can also give decompositions of 1-qubit gates in terms of rotations about any other two non-parallel axes of the Bloch sphere.

Theorem IV-2 : [34]

Suppose U is a 1-qubit unitary gate. Let l and m be any two non-parallel axes of the Bloch sphere. Then there exist real numbers α, β, γ, and δ such that

$$U = e^{i\alpha}R_l(\beta)R_m(\gamma)R_l(\delta) \quad \ldots \qquad \textbf{(IV-11)}$$

As a result any 1-qubit gate U can be written in the form

$$U = e^{i\alpha}AXBXC, \ldots \qquad \textbf{(IV-12)}$$

where A,B,C are unitary operators satisfying ABC = I. (Recall that the Pauli gate X is the NOT gate).

IV-3-1-2- Controlled-*U* Gates

A *controlled*-not (CNOT) gate is a 2-qubit quantum gate that conditionally applies the not gate on the second (target) qubit when the first (control qubit) is in state $|1\rangle$. Remember that such a gate acts on quantum states in quantum superposition.

Given any 1-qubit gate *U*, we can similarly define a *controlled-U* gate, denotedc-*U*, which will be a 2-qubit gate corresponding to the following operation:

$$\begin{aligned} c-U|0\rangle|\psi\rangle &= |0\rangle|\psi\rangle \\ c-U|1\rangle|\psi\rangle &= |1\rangle U|\psi\rangle \end{aligned} \quad \ldots \qquad \textbf{(IV-13)}$$

The symbol commonly used for the c-*U* gate in a quantum circuit diagram is shown in ***Figure IV-5***.

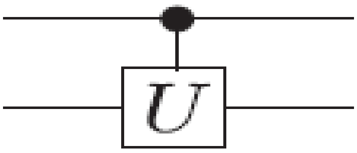


Figure IV-5: The c-*U* gate. [34]

The construction of a controlled-*U* for any 1-qubit gate *U* can be generalized to allow the implementation of a controlled version of any quantum circuit implementing a unitary operation *U*. Suppose we are given a circuit C_U implementing a unitary *U*, and we wish to implement a circuit for the controlled-*U* operation. The basic technique is to replace every gate *G* in C_U by a controlled gate c-*G*, as shown in ***Figure V-6***.

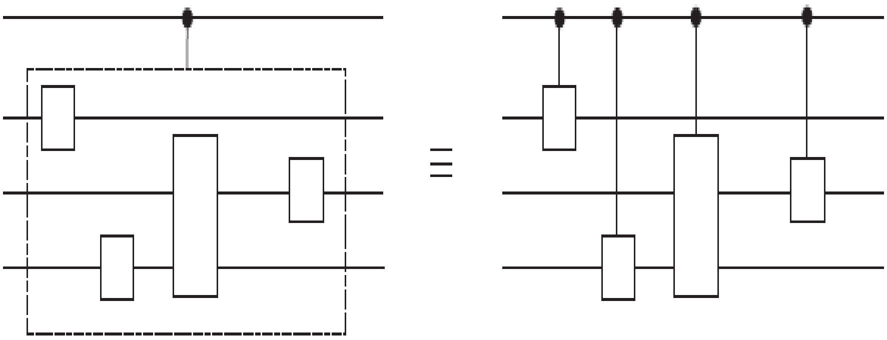


Figure IV-6: Given a circuit *CU* implementing a unitary *U. [34]*

We can assume without loss of generality that C_U consists only of 1-qubit gates and CNOT gates. So the only thing that remains is to construct a controlled version of the CNOT gate. By the way a controlled-cnot gate is called a Toffoli gate. The Toffoli gate can be implemented by a circuit containing cnot gates and some 1-qubit gates. So we can use this replacement for each of the Toffoli gates generated in our construction of the controlled-*U* circuit. This completes the construction of a circuit for implementing the controlled-*U* operation.

IV-3-2- Universal Sets of Quantum Gates:

The gates we have seen so far have acted on either a single qubit, or on two qubits. An interesting quantum algorithm would, in general, be some complicated unitary operator acting non-trivially on *n*-qubits. In classical computing, we implement complicated operations as a sequence of much simpler operations. In practice, we want to be able to select these simple operations from some set of elementary gates. In quantum computing, we do the same thing. The goal is to choose some finite set of gates so that, by constructing a circuit using only gates from that set, we can implement non-trivial and interesting quantum computations.

When we use a circuit of quantum gates to implement some desired unitary operation, in practice, it suffices to have an implementation that *approximates* the desired unitary to some specified level of accuracy. We need to make precise the notion of the *quality of an approximation* of a unitary transformation. Suppose we approximate a desired unitary transformation U by some other unitary transformation V . The *error* in the approximation is defined to be

$$E(U,V) \equiv \max_{|\psi\rangle} \left\| (U - V)|\psi\rangle \right\| \quad \ldots \qquad \textbf{\textit{(IV-14)}}$$

When we say that an operator U can be 'approximated to arbitrary accuracy', we mean that if we are given any error tolerance $\varepsilon > 0$, we can implement some unitary V such that $E(U, V) < \varepsilon$. Having

$$E(U_2U_1, V_2V_1) \leq E(U_2, V_2) + E(U_1, V_1) \quad \ldots \qquad \textbf{\textit{(IV-15)}}$$

It follows that

$$E(U_nU_{n-1} \ldots U_1, V_nV_{n-1} \ldots V_1) \leq E(U_n, V_n) + E(U_{n-1}, V_{n-1}) + \cdots + E(U_1, V_1) \ldots \textbf{\textit{(IV-16)}}$$

Definition IV-1:

A set of gates is said to be universal *if for any integer $n \geq 1$, any n-qubit unitary operator can be approximated to arbitrary accuracy by a quantum circuit using only gates from that set.*

Finding convenient universal sets of gates is of great practical importance as well as of theoretical interest. Since a universal set of gates must be able to implement, for example, the CNOT, it will have to contain at least one non-trivial gate on two or more qubits.

Definition IV-2:

A 2-qubit gate is said to be an entangling gate *if for some input product state $|\psi\rangle|\phi\rangle$ the output of the gate is not a product state (i.e. the output qubits are entangled).*

The following universality result is a useful starting point.

Theorem IV-3:[35]

A set composed of any 2-qubit entangling gate, together with all 1-qubit gates, is universal.

Theorem V-3 implies, for example, that the CNOT gate together with all 1-qubit gates is universal. The theorem gives sets that are universal in a stronger sense required by ***Definition IV-1***. With an entangling 2-qubit gate and all 1-qubit gates, we can implement any *n*-qubit unitary *exactly*. A shortcoming of ***Theorem IV-3*** is that the universal sets of gates it provides are infinite. It is useful to find a *finite* set of gates that is universal. A natural starting point in this direction is to look for a finite set of 1-qubit gates that can be used to approximate any 1-qubit gate to arbitrary accuracy.

Definition IV-3:

A set of gates is said to be universal for 1-qubit gates *if any 1-qubit unitary gate can be approximated to arbitrary accuracy by a quantum circuit using only gates from that set.*

Theorem IV-2 states that for any two non-parallel axes *l* and *m* of the Bloch sphere, the set consisting of the rotation gates *Rl*(β) and *Rm*(γ) for all $\beta, \gamma \in [0, 2\pi]$ is universal for 1-qubit gates.

Theorem IV-4:[35]

If a set of two 1-qubit gates (rotations) $G = \{R_l(\beta), R_m(\gamma)\}$ *satisfies the conditions:*

1- *l and m are non-parallel axes of the Bloch sphere*

2- $\beta, \gamma \in [0, 2\pi]$ *are real numbers such that* β/π *and* γ/π *are not rational then G is universal for 1-qubit gates.*

As a concrete example, we give a simple set satisfying the conditions of ***Theorem IV-4***. In this direction is the *Hadamard* gate, *H*, and the $\frac{\pi}{8}$-phase gate, T, where

$$H = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad and \quad T = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{4}} \end{bmatrix} \qquad \ldots \qquad \textbf{\textit{(IV-17)}}$$

The set G = {HTHT,THTH} satisfies the conditions of **Theorem IV-4**, *this gives:*

The set {H, T} is universal for 1-qubit gates. We now have the following universality result.

Theorem IV-5:[35]

*The set {*CNOT,*H, T} is a universal set of gates.*

IV-3-3- Efficiency of Approximating Unitary Transformations

In the previous section, we have stated that an arbitrary unitary transformation can be simulated using gates from a fixed universal set, such as *{H,* CNOT, *T}**(Theorem IV-5)***.** We have said nothing about how *efficiently* this can be done however. If we wish to implement a given unitary transformation *U* (corresponding to some computation), we would be interested in being able to do this using a *polynomial* number of gates from our universal set. Here, 'polynomial' is taken to mean 'polynomial in $\frac{1}{\varsigma}$ *and* in the number of qubits *n*', where ς is the desired quality of the estimate of *U*.

In fact, *most* unitary transformations *cannot* be *efficiently* approximated using gates from our universal set; this can be shown by counting arguments (since there are many more transformations than efficient circuits).

The difficulty in efficiently implementing some unitary transformations does not lie in the complexity of simulating arbitrary 1-qubit gates from a finite set of 1-qubit gates, since the decomposition described before can be done in time polynomial in $\frac{1}{\varsigma}$ provided *n*-bit approximations of all the coefficients of the gates can be computed in time polynomial in *n*. A result known as the *Solovay–Kitaev theorem* promises that we can do much better and find a set *G* of 1-qubit gates such that any arbitrary 1-qubit gate can be approximated to arbitrary accuracy using a sequence of a poly-logarithmic number of gates from *G*. In other words, if we want to approximate a given unitary with error less than ε, we can do so using a number of gates that is polynomial in log(1/ε).

It is worth discussing some of the consequences of the Solovay–Kitaev theorem. Suppose we are given a quantum circuit consisting of several CNOT gates, and *m* 1-qubit gates, and we wish to approximate this circuit using only gates from the universal set {CNOT} $\in$ *G*. Suppose we approximate each 1-qubit gate in the circuit with error at most $\frac{\varsigma}{m}$. Then the overall error in the approximation of the circuit is bounded by ς. So, if we want to approximate the circuit using only gates from our universal set {CNOT} $\in$ *G*, and if we want the total error in the approximation to be at most ς, we should aim to approximate each 1-qubit gate in the circuit with error at $\frac{\varsigma}{m}$. We are now faced with the following question of efficiency: 'how many gates from G are required to approximate each 1-qubit gate with error at most $\frac{\varsigma}{m}$?' A special case of the Solovay–Kitaev theorem answers this question.

Theorem IV-6 [35]

(Solovay–Kitaev) If G is a finite set of 1-qubit gates satisfying the conditions of ***Theorem IV-4*** *and also*

3- *for any gate $g \in G$, its inverse g^{-1} can be implemented exactly by a finite sequence of gates in G, then any 1-qubit gate can be approximated with error at most ς using $O\log^{c}\left(\frac{1}{\varsigma}\right)$ gates from G, where c is a positive constant.*

Thus, according to the Solovay–Kitaev theorem, any 1-qubit gate can be approximated with error at most $\frac{\varsigma}{m}$ using $O\log^{c}\left(\frac{1}{\varsigma}\right)$ gates from a finite set G that is universal for 1-qubit gates, and that contains its own inverses (or whose inverses can be constructed exactly from a finite sequence of gates from G). It is worth noting that if n-bit approximations of the coefficients of the gates in G can be computed in time polynomial in n, then the efficient decompositions can be found in time polynomial in $\log\left(\frac{1}{\varsigma}\right)$.

Notice that the set *{H, T}* satisfies these conditions. For a circuit having m
1-qubit gates, the approximation of these gates requires at most

$$O(m\log^{c}\left(\frac{m}{\varsigma}\right)) \qquad \ldots \qquad \textbf{\textit{(IV-18)}}$$

gates from a universal set. This is a poly-logarithmic increase over the size of the original circuit.

IV-3-4- Implementing Measurements with Quantum Circuits

Given an orthonormal basis $\left|\varphi_j\right\rangle$, suppose we have a state $\left|\psi\right\rangle$, which we write in this basis:

$$\left|\psi\right\rangle = \sum_j \alpha_j \left|\varphi_j\right\rangle \qquad \ldots \qquad \textbf{\textit{(IV-19)}}$$

Recall that a Von Neumann measurement of $\left|\psi\right\rangle$ with respect to the basis $\left\{\left|\varphi_j\right\rangle\right\}$is described by the orthogonal projectors $\left\{\left|\varphi_j\right\rangle\left\langle\varphi_j\right|\right\}$, and will output the result '*j*' with probability

$$Tr\left(\left|\psi\right\rangle\left\langle\psi\right\|\varphi_j\right\rangle\left\langle\varphi_j\right|\right) = \left|\alpha_j\right|^2 \qquad \ldots \qquad \textbf{\textit{(IV-20)}}$$

Given a device that will measure individual qubits in the computational basis, we can use a quantum circuit to implement Von Neumann measurements of a multi-qubit register with respect to any orthonormal basis $\left\{\left|\varphi_j\right\rangle\right\}$. This can be done as follows. First, we construct a quantum circuit that implements the unitary transformation $U\left|\varphi_j\right\rangle = \left|j\right\rangle$, where $\left|j\right\rangle$is the corresponding *n*-qubit computational basis state). The operator *U* performs a basis change from the $\left\{\left|\varphi_j\right\rangle\right\}$basis to the computational basis. Given a general state $\sum_j \alpha_j \left|\varphi_j\right\rangle$, we use the circuit to perform the basis change *U*, and then make a measurement of the register in the computational basis. Finally, we perform the inverse basis change U^{-1} (by running the circuit for *U* backwards, replacing each gate by its inverse). This network is shown in ***Figure IV-7***. An alternative approach is illustrated in ***Figure IV-8***. In the alternative approach, we do not directly measure the state (with respect to the computational basis) after the basis change, but instead we "copy" the values onto an ancillary register, which we then measure in the computational basis.

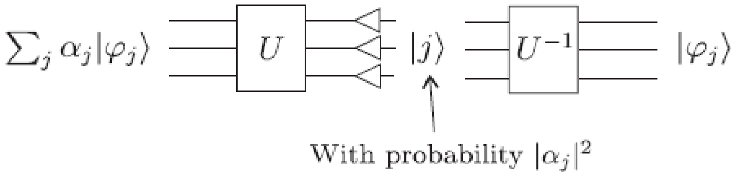

$\sum_j \alpha_j|\varphi_j\rangle$
U
$|j\rangle$
U^{-1}
$|\varphi_j\rangle$
With probability $|\alpha_j|^2$

Figure IV-7: Circuit implementing a Von Neumann measurement with respect to the basis $\left\{\left|\varphi_j\right\rangle\right\}$. [34]

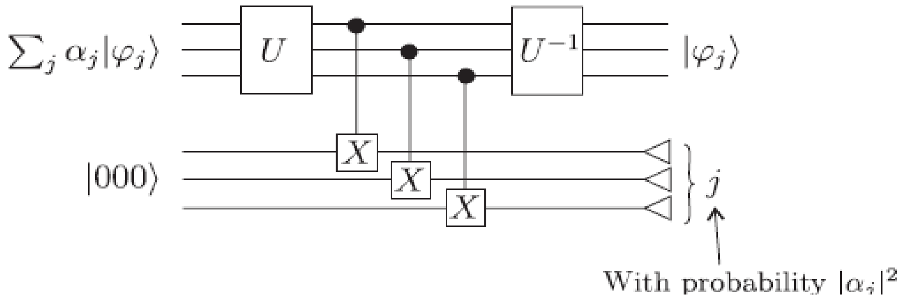

$\sum_j \alpha_j|\varphi_j\rangle$
U
U^{-1}
$|\varphi_j\rangle$
$|000\rangle$
X
X
X
j
With probability $|\alpha_j|^2$

Figure IV-8: Another circuit implementing the Von Neumann measurement. [34]

It will be very important for quantum computing to be able to implement general projective measurements, and not complete Von Neumann measurements. Consider a projective measurement with respect to the decomposition

$$I = \sum_i P_i \qquad \ldots \qquad \textit{(IV-21)}$$

where P_i has rank r_i. In other words

$$P_i = \sum_{j=1}^{r} |\psi_{i,j}\rangle\langle\psi_{i,j}| \qquad \ldots \qquad \textit{(IV-22)}$$

where the the states $\{|\psi_{i,j}\rangle\}$are an orthonormal basis for the Hilbert space of dimension $N = \sum_i r_i$.

Let U_P be a circuit that maps$|\psi_{i,j}\rangle|0\rangle \mapsto |\psi_{i,j}\rangle|i\rangle$. One way (but not the only way) to implement U_P is to perform a basis change$U : |\psi_{i,j}\rangle \mapsto |i,j\rangle$, 'copy' j to the ancilla register, and then apply U^{-1}.

One can implement U_P with a sequence of CNOT gates, as illustrated in ***Figure IV-9***. Thus after the U_P circuit, we have the state

$$\sum_x \alpha_x |x\rangle| \, parity(x)\rangle = \sum_{parity(x)=0} \alpha_x |x\rangle|0\rangle + \sum_{parity(x)=1} \alpha_x |x\rangle|1\rangle = \alpha_0|\psi_0\rangle|0\rangle + \alpha_1|\psi_1\rangle|1\rangle \quad \ldots \textit{(IV-23)}$$

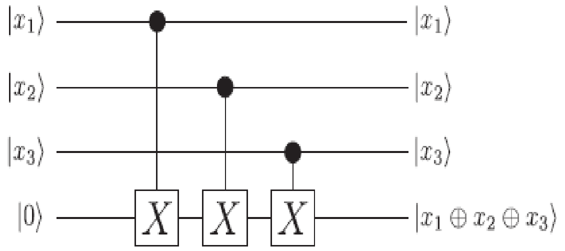


Figure IV-9: A circuit computing the parity of three qubits. [34]

Thus measuring the ancilla qubit will leave the first register in the state $|\psi_0\rangle$with probability $|\alpha_0|^2$ and in the state $|\psi_1\rangle$with probability $|\alpha_1|^2$, as required. Therefore, this circuit will implement a parity measurement on an arbitrary 3- qubit state, as depicted in ***Figure IV-10***.

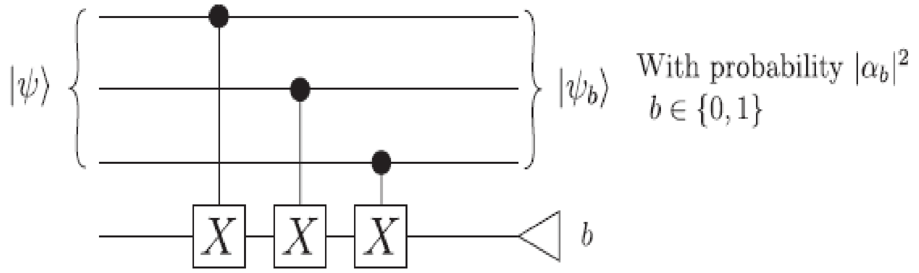


Figure IV-10: A circuit implementing a parity measurement. [34]

It is worth emphasizing what differentiates this projective parity measurement from a Von Neumann measurement followed by classical post-processing to compute the parity. The projective measurement measures *only* the parity of the strings in the quantum state, and no other information, leaving one of the superposition states $|\psi_0\rangle$or$|\psi_1\rangle$. A complete Von Neumann measurement would have extracted more information than needed, and we would have been left with a random basis state $|x\rangle$of a specific parity instead of a superposition of all strings with the same parity.

IV-4- The programation process of a quantum computer

The good thing about the classical computer is that it is programmable. It is a single device capable of performing different operations depending on the program it is given: word processing, algebraic transformations, displaying movies, etc.. To put it in more abstract words a classical computer is a universal gate array: we can program every possible function with n input and n output bits by specifying a program of length $n2^n$. That is, a fixed circuit with $n(1 + 2^n)$ input bits can be used in order to compute any function on the first n bits in the register. Is the same true for quantum computers? Or will these devices typically be made-to-measure with respect to a single task?

Nielsen and Chuang [34] showed that quantum computers cannot be universal gate arrays. Even if the program is itself given in form of a quantum state it would require a program register of infinite length in order to perform an arbitrary (unitary) operation on a finite number of qubits – universality was shown to be only possible in a probabilistic manner. In this sense, quantum computers will not be the kind of all purpose devices which classical computers are. In practice, however, any finite set of quantum programs can run on a quantum computer with a finite program register. This issue applies, however, to the programming of a quantum computer with a fixed hardware, which is, needless to say, still in the remote future as a physical device.

IV-5- Elementary quantum algorithms

In the same scientific paper in which David Deutsch [44] introduced the notion of the universal quantum computer, he also presented the first quantum algorithm. The problem that this algorithm addresses, later referred to as Deutsch's problem, is a very simple one. Yet the Deutsch algorithm already exemplifies the advantages of a quantum computer through skilfully exploiting quantum parallelism. Like the Deutsch algorithm, all other elementary quantum algorithms amount to deciding which black box out of finitely many alternatives one has at hand. Such a black box is often also referred to as oracle. An input may be given to the oracle, one may read out or use the outcome in later steps of the quantum algorithm, and the objective is to find out the functioning of the black box. It is assumed that this oracle operation can be implemented with some sequence of quantum logic gates. The complexity of the quantum algorithm is then quantified in terms of the number of queries to the oracle.

IV-5-1- Probabilistic Versus Quantum Algorithms

We begin by considering a simple probabilistic computation. ***Figure IV-11*** illustrates the first two steps of such a computation on a register that can be in one of the four states, labelled by the integers 0, 1, 2, and 3. Initially the register is in the state 0. After the first step of the computation, the register is in the state j with probability $p_{0,j}$. For example, the probability that the computation is in state 2 after the first step is $p_{0,2}$. In the second step of the computation, the register goes from state j to state k with probability $q_{j,k}$. For example, in the second step the computation proceeds from state 2 to state 3 with probability $q_{2,3}$.

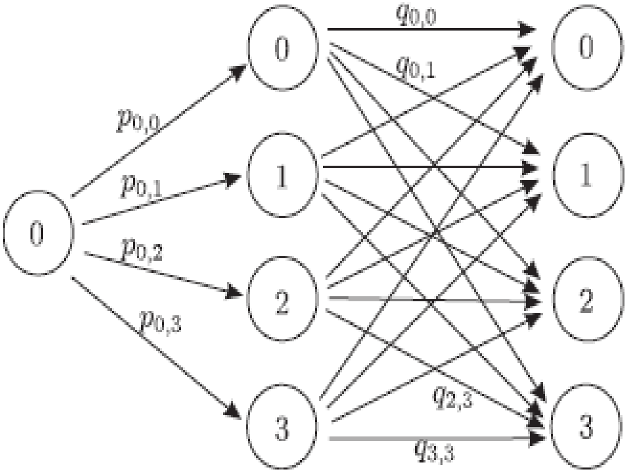


Figure IV-11: A classical probabilistic computation acting on a register that can be in one of four states labelled 0, 1, 2, 3. [23]

Suppose we want to find the total probability that the computation ends up in state 3 after the second step. This is calculated by first determining the probability associated with each computation 'path' that could end up at the state 3, and then by adding the probabilities for all such paths. There are four computation paths that can leave the computation in state 3 after the first step. The computation can proceed from state 0 to state *j* and then from state *j* to state 3, for any of the four $j \in \{0,1,2,3\}$. The probability associated with any one of these paths is obtained by multiplying the probability $p_{0,j}$ of the transition from state 0 to state *j*, with the probability $q_{j,3}$ of the transition from state *j* to state 3. The total probability of the computation ending up in state 3 is given by adding these four possibilities. So we have

$$\text{prob(final outcome is 3)} = \sum_j p_{0,j} q_{j,3} \qquad \dots \qquad \textbf{\textit{(IV-24)}}$$

Another way of looking at this computation is to suppose the register consists of two qubits, and let the labels 0, 1, 2, 3 refer to the four basis states $|00\rangle, |01\rangle, |10\rangle, |11\rangle$, respectively. Then view each of the transition probabilities as a squared norm of a quantum probability amplitude, so that $p_{0,j} = |\alpha_{0,j}|^2$ and $q_{j,k} = |\beta_{j,k}|^2$. This approach is shown in ***Figure IV-12***, which can be viewed as a quantum computation in which the state is measured after each step.

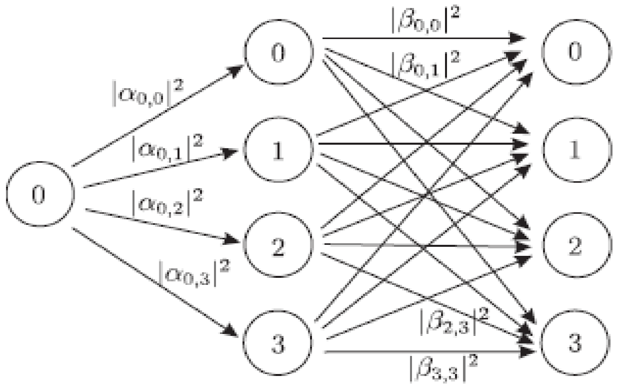


Figure IV-12**:** The classical probabilistic computation viewed in a quantum setting. [23]

As before, the total probability of measuring outcome 3 after the second step is

$$\text{prob(final outcome is 3)} = \sum_{j} |\alpha_{0,j}|^2|\beta_{j,3}|^2 = \sum_{j} |\alpha_{0,j}\beta_{j,3}|^2 \quad \ldots \qquad \textbf{\textit{(IV-25)}}$$

which is the same probability as in ***Equation (IV-24)***.

In this example, since we assume that the state is measured after each step, we would know the intermediate state *j*, and thus we would know which computation path leading to the final state 3 was taken. The total probability of arriving at the final state 3 is determined by adding the squared norm of the probability amplitude $\alpha_{0,j}\beta_{j,3}$ associated with each path (i.e. we add the *probabilities* for the four paths, and not the probability amplitudes).

In a fully quantum algorithm, we would not measure the state immediately after the first step. This way the quantum probability amplitudes will have a chance to *interfere*. For example, some negative amplitudes could cancel with some positive amplitudes, significantly affecting the final probabilities associated with a given outcome. A quantum version of the algorithm above is illustrated in ***Figure IV-13***.

This time the calculation of the total probability associated with outcome 3 in the measurement after the second step is different. Since there is no measurement after the first step of the computation, we do not learn the path taken by the computation to the final state 3. That is, when we obtain the output 3, we will have no information telling us which of the four paths was taken. In this case, instead of adding the probabilities associated with each of these four paths, we must add the probability *amplitudes*. The probability of a measurement after the second step giving the result 3 is obtained by taking the squared norm of the *total probability amplitude*.

$$\text{prob(final outcome is 3)} = \left|\sum_j \alpha_{0,j}\beta_{j,3}\right|^2 \quad \ldots \quad \textbf{\textit{(IV-26)}}$$

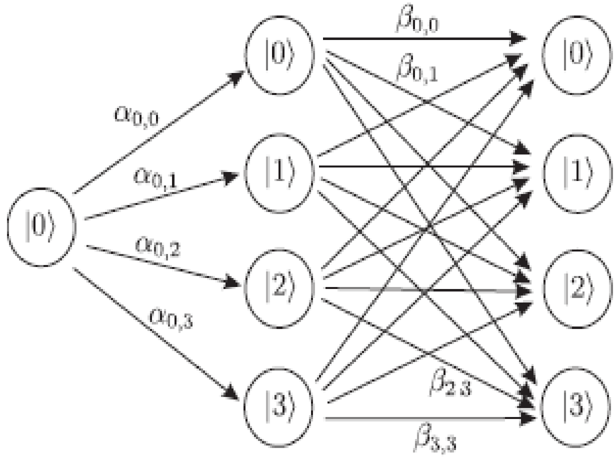


Figure IV-13: A fully quantum computation. [23]

Which is clearly distinct from the classical result; ***Equation (IV-24)***

Now consider the quantum circuit in ***Figure IV-14***. This circuit does not perform a purely quantum computation, because we make a measurement immediately after the first Hadamard gate.

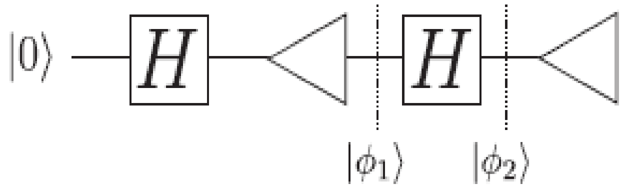


Figure IV-14: A quantum circuit exhibiting no quantum interference. [23, 36]

The state $|\phi_1\rangle$ immediately after this measurement is

$$|\phi_1\rangle = \begin{cases} |0\rangle \textit{with} \quad \textit{probability}\,\frac{1}{2} \\ |1\rangle \textit{with} \quad \textit{probability}\,\frac{1}{2} \end{cases} \quad \ldots \quad \textbf{\textit{(IV-27)}}$$

The state immediately after the second Hadamard gate is then

$$|\phi_2\rangle = \begin{cases} \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right) with \quad probability \;\; \frac{1}{2} \\ \frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right) with \quad probability \;\; \frac{1}{2} \end{cases} \qquad \ldots \qquad \textbf{\textit{(IV-28)}}$$

In either case, the final measurement will give the result 0 or 1 with equal probability. Compare the above with the quantum circuit shown in ***Figure IV-15***. This time there is no measurement after the first Hadamard gate, and the application of the second Hadamard gate will give rise to interference in the quantum amplitudes. The state immediately after the first Hadamard gate is

$$|\psi_1\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle \qquad \ldots \qquad \textbf{\textit{(IV-29)}}$$

This state is input directly to the second Hadamard gate, and the state after the second Hadamard gate is

$$\begin{aligned} |\psi_2\rangle &= H\left(\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle\right) \\ &= \frac{1}{\sqrt{2}}H|0\rangle + \frac{1}{\sqrt{2}}H|1\rangle \\ &= \frac{1}{\sqrt{2}}\left(\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle\right) + \frac{1}{\sqrt{2}}\left(\frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle\right) \qquad \ldots \qquad \textbf{\textit{(IV-30)}} \\ &= \frac{1}{2}|0\rangle + \frac{1}{2}|1\rangle + \frac{1}{2}|0\rangle - \frac{1}{2}|1\rangle \\ &= |0\rangle \end{aligned}$$

The total probability amplitude associated with $|1\rangle$ is 0, meaning that the probability for the second measurement giving result ‘1’ is now 0. The second Hadamard gate acted on the basis states $|0\rangle$ and $|1\rangle$ in superposition, and the amplitudes of state $|1\rangle$ for the two paths in this superposition interfered, causing them to cancel out.

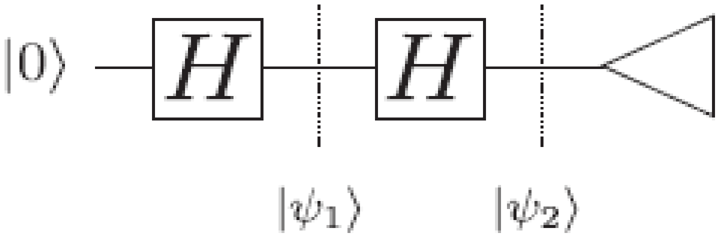


Figure IV-15: A quantum circuit exhibiting interference. [23, 24]

IV-5-2- Deutsch algorithm

The Deutsch algorithm is a very simple example of a quantum algorithm based on the Quantum Fouries Transform it illustrates the key ideas of *quantum parallelism* and *quantum interference* that are used in all useful quantum algorithms.

The problem solved by the Deutsch algorithm is the following. Suppose we are given a reversible circuit for computing an unknown 1-bit function $f:\{0,1\}\mapsto\{0,1\}$. We treat this reversible circuit as a 'black box' or 'oracle'. This means that we can apply the circuit to obtain values of $f(x)$ for given inputs x, but we cannot gain any information about the inner workings of the circuit to learn about the function f. The problem is to determine the value of $f(0) \oplus f(1)$. If we determine that $f(0) \oplus f(1) = 0$, then we know that $f(0) = f(1)$ (although we do not know the value), and we say that f is 'constant'. If on the other hand we determine that $f(0) \oplus f(1) = 1$, then we know that $f(0) \neq f(1)$, and we say the function is 'balanced'. So determining $f(0) \oplus f(1)$ is equivalent to determining whether the function f is constant or balanced.

How many queries to the oracle for f must be made classically to determine $f(0) \oplus f(1)$? Clearly the answer is 2. Suppose we compute $f(0)$ using one (classical) query. Then the value of $f(1)$ could be 0, making $f(0) \oplus f(1) = 0$, or the value of $f(1)$ could be 1, making $f(0) \oplus f(1) = 1$. Without making a second query to the oracle to determine the value of $f(1)$, we can make no conclusion about the value of $f(0) \oplus f(1)$. The Deutsch algorithm is a quantum algorithm capable of determining the value of $f(0) \oplus f(1)$ by making only a *single query* to a quantum oracle for f.

The given reversible circuit for f can be made into a quantum circuit, by replacing every reversible classical gate in the given circuit with the analogous unitary quantum gate. This quantum circuit can be expressed as a unitary operator

$U_f : |x\rangle|y\rangle \mapsto |x\rangle|y \oplus f(x)\rangle$

Having created a quantum version of the circuit for f, we can supply *quantum bits* as inputs. We define U_f so that if we set the second input qubit to be in the state $|y\rangle = |0\rangle$, then $|x\rangle = |0\rangle$ in the first input qubit will give $|0 \oplus f(0)\rangle = |f(0)\rangle$ in the second output bit, and $|x\rangle = |1\rangle$ in the first input qubit will give $|f(1)\rangle$. So we can think of $|x\rangle = |0\rangle$ as a quantum version of the (classical) input bit 0, and $|x\rangle = |1\rangle$ as a quantum version of the input bit 1. Of course, the state of the input qubit can be some *superposition* of $|0\rangle$ and $|1\rangle$. Suppose, still keeping the second input qubit $|y\rangle = |0\rangle$, we set the first input qubit to be in the superposition state

$$\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle \qquad \dots \qquad \textbf{\textit{(IV-31)}}$$

Then the two qubit input to U_f is

$$\begin{aligned}&(\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle)|0\rangle \\ &= \frac{1}{\sqrt{2}}|0\rangle|0\rangle + \frac{1}{\sqrt{2}}|1\rangle|0\rangle\end{aligned} \qquad \dots \qquad \textbf{\textit{(IV-32)}}$$

The output of U_f will be the state

$$U_f\,(\frac{1}{\sqrt{2}}|0\rangle|0\rangle + \frac{1}{\sqrt{2}}|1\rangle|0\rangle)$$

$$= \frac{1}{\sqrt{2}}U_f\,|0\rangle|0\rangle + \frac{1}{\sqrt{2}}U_f\,|1\rangle|0\rangle$$

$$= \frac{1}{\sqrt{2}}|0\rangle|0 \oplus f(0)\rangle + \frac{1}{\sqrt{2}}|1\rangle|0 \oplus f(1)\rangle$$

$$= \frac{1}{\sqrt{2}}|0\rangle|f(0)\rangle + \frac{1}{\sqrt{2}}|1\rangle|0 \oplus f(1)\rangle$$

In some sense, U_f has *simultaneously* computed the value of f on both possible inputs 0 and 1 in superposition. However, if we now measure the output state in the computational basis, we will observe *either* $|0\rangle|f(0)\rangle$ (with probability 1/2), *or* $|1\rangle|0 \oplus f(1)\rangle$ (with probability 1/2). After the measurement, the output state will be either $|f(0)\rangle$ or $|f(1)\rangle$ respectively, and so any subsequent measurements of the output state will yield the same result. So this means that although we have successfully computed two values in superposition, only one of those values is accessible through a quantum measurement in the computational basis. Fortunately, this is not the end of the story.

Recall that for the Deutsch problem we are ultimately not interested in individual values of $f(x)$, but wish to determine the value of $f(0) \oplus f(1)$. The Deutsch algorithm illustrates how we can use *quantum interference* to obtain such *global information* about the function *f*, and how this can be done more efficiently than is possible classically. The Deutsch algorithm is implemented by the quantum circuit shown in ***Figure IV-16***.

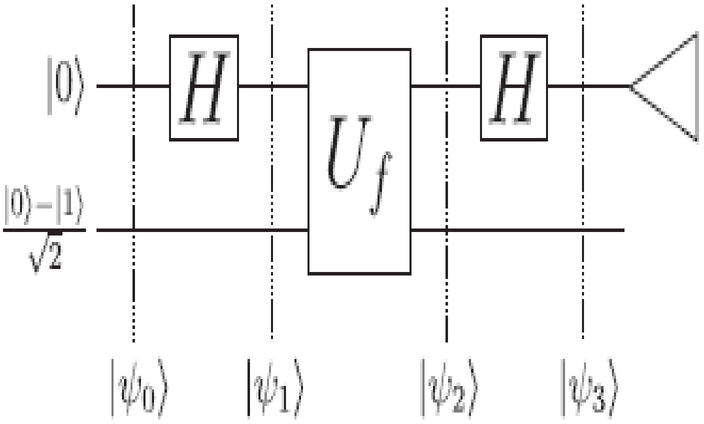


Figure IV-16: A circuit implementing the Deutsch algorithm. The measured value equals *f*(0) $\oplus$ *f*(1). [37]

Note that the second input bit has been initialized to the state $\frac{|0\rangle-|1\rangle}{\sqrt{2}}$. This state can easily be created from the state $|1\rangle$ by applying a single Hadamard gate. We do not show this gate, however, to emphasize a certain symmetry that is characteristic of these algorithms. A convenient way to analyse the behaviour of a quantum algorithm is to work through the state at each stage of the circuit. First, the input state is

$$|\psi_0\rangle = |0\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right) \qquad \ldots \qquad \textbf{\textit{(IV-33)}}$$

After the first Hadamard gate is applied to the first qubit, the state becomes

$$\begin{aligned}|\psi_1\rangle &= \left(\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle\right)\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right) \\ &= \frac{1}{\sqrt{2}}|0\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right) + \frac{1}{\sqrt{2}}|1\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)\end{aligned} \qquad \ldots \qquad \textbf{\textit{(IV-34)}}$$

Recalling from Appendix A-3, after applying the U_f gate we have the state

$$\begin{aligned}\psi_2 &= \frac{(-1)^{f(0)}}{\sqrt{2}}|0\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)+\frac{(-1)^{f(1)}}{\sqrt{2}}|1\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)\\ &=\left(\frac{(-1)^{f(0)}|0\rangle+(-1)^{f(1)}|1\rangle}{\sqrt{2}}\right)\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right) \qquad \dots \qquad \textbf{\textit{(IV-35)}}\\ &=(-1)^{f(0)}\left(\frac{|0\rangle+(-1)^{f(0)\oplus f(1)}|1\rangle}{\sqrt{2}}\right)\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)\end{aligned}$$

where the last equality uses the fact that $(-1)^{f(0)}(-1)^{f(1)} = (-1)^{f(0)\ \oplus f(1)}$. If f is a constant function (i.e. $f(0) \oplus f(1) = 0$), then we have

$$\psi_2 = (-1)^{f(0)}\left(\frac{|0\rangle+|1\rangle}{\sqrt{2}}\right)\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right) \qquad \dots \qquad \textbf{\textit{(IV-36)}}$$

and so the final Hadamard gate on the first qubit transforms the state to

$$\psi_3 = (-1)^{f(0)}|0\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right) \qquad \dots \qquad \textbf{\textit{(IV-37)}}$$

The squared norm of the basis state $|0\rangle$ in the first qubit is 1. This means that for a constant function a measurement of the first qubit is certain to return the value $0 = f(0) \oplus f(1)$.

If f is a balanced function (i.e. $f(0) \oplus f(1) = 1$), then we have

$$\psi_2 = (-1)^{f(0)}\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right) \qquad \dots \qquad \textbf{\textit{(IV-38)}}$$

and so the final Hadamard gate on the first qubit transforms the state to

$$\psi_3 = (-1)^{f(0)}|1\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right) \qquad \dots \qquad \textbf{\textit{(IV-39)}}$$

In this case the squared norm of the basis state $|1\rangle$ in the first qubit is 1. This means that for a balanced function a measurement of the first qubit is certain to return the value $1 = f(0) \oplus f(1)$. So a measurement of the first qubit at the end of the circuit for the Deutsch algorithm determines the value $f(0) \oplus f(1)$ and thus whether the function is constant or balanced.

To gain some insight into how the Deutsch algorithm can generalize, it is helpful to remember that the operator $U_f : |x\rangle|y\rangle \mapsto |x\rangle|y \oplus f(x)\rangle$ in the Deutsch algorithm can be viewed as a single-qubit operator $\hat{U}_{f(x)}$, whose action on the second qubit is controlled by the state of the first qubit ***(Figure IV-17)***. The state $\left(\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right)$ is an eigenstate of $\hat{U}_{f(x)}$ with eigenvalue $(-1)^{f(x)}$. By encoding these eigenvalues in the phase factors of the control qubit, we are able to determine $f(0) \oplus f(1)$ by determining the relative phase factor between $|0\rangle$ and $|1\rangle$. Distinguishing $\left(\frac{|0\rangle + |1\rangle}{\sqrt{2}}\right)$ and $\left(\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right)$ is done using the Hadamard gate.

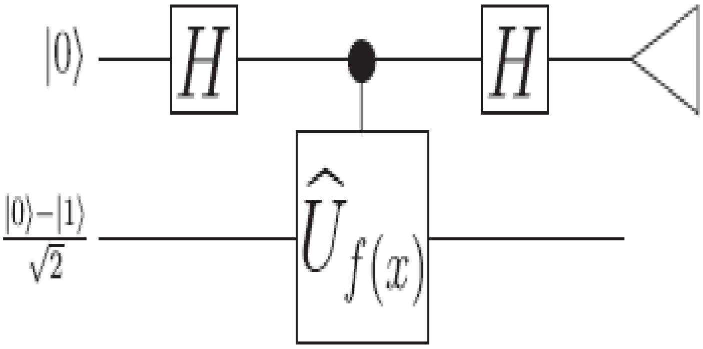


Figure IV-17: The circuit for Deutsch's algorithm with the $c-U_{f(x)}$ drawn instead of *U_f [37]*

IV-5-3- Deutsch-Jozsa algorithm [44]

The Deutsch–Jozsa algorithm solves a problem that is a straight forward generalization of the problem solved by the Deutsch algorithm. The algorithm has exactly the same structure. As with the Deutsch algorithm, we are given a reversible circuit implementing an unknown function *f*, but this time *f* is a function from *n*-bit strings to a single bit. That is,

$$f : \{0,1\}^n \rightarrow \{0,1\} \quad \ldots \quad \textbf{\textit{(IV-40)}}$$

We are also given the *promise* that *f* is either *constant* (meaning *f*(*x*) is the same for all *x*), or *f* is *balanced* (meaning *f*(*x*) = 0 for exactly half of the input strings *x*, and *f*(*x*) = 1 for the other half of the inputs). The problem here is to determine whether *f* is constant, or balanced, by making queries to the circuit for *f*.

Consider solving this problem by a classical algorithm. Suppose we have used the oracle to determine *f*(*x*) for exactly half of the possible inputs *x* (i.e. you have made 2^{n-1} queries to *f*), and that all queries have returned *f*(*x*) = 0. At this point, we would strongly suspect that *f* is constant. However, it is possible that if we queried *f* on the remaining 2^{n-1} inputs, we might get $f(x) = 1$ each time. So it is still possible that f is balanced. So in the worst case, using a classical algorithm we cannot decide with certainty whether f is constant or balanced using any less than $2^{n-1}+1$ queries. The property of being constant or balanced is a global property of f. As for the Deutsch problem, a quantum algorithm can take advantage of quantum superposition and interference to determine this global property of f. The Deutsch–Jozsa algorithm will determine whether f is constant, or balanced, making only *one* query to a quantum version of the reversible circuit for f.

Analogous to what we did for the Deutsch algorithm, we will define the quantum operation

$$U_f : |X\rangle|y\rangle \mapsto |X\rangle|y \oplus f(x)\rangle \qquad \ldots \qquad \textbf{\textit{(IV-41)}}$$

This time we write **x** in boldface, because it refers to an n-bit string. As before, we think of U_f as a 1-qubit operator$\hat{U}_{f(x)}$, this time controlled by the *register* of qubits in the state$|x\rangle$. We can see that $\frac{|0\rangle - |1\rangle}{\sqrt{2}}$ is an eigenstate of $\hat{U}_{f(x)}$with eigenvalue $(-1)^{f(\mathbf{x})}$.

The circuit for the Deutsch–Jozsa algorithm is shown in ***Figure IV-18***. Notice the similarity between the circuit for the Deutsch algorithm, and the circuit for the Deutsch–Jozsa algorithm. In place of a simple 1-qubit Hadamard gate, we now have tensor products of n 1-qubit Hadamard gates (acting in parallel).

This is denoted $H^{\otimes n.}$, We use$|0\rangle^{\otimes n}$, or $|0\rangle$ to denote the state that is the tensor product of n qubits, each in the state $|0\rangle$.

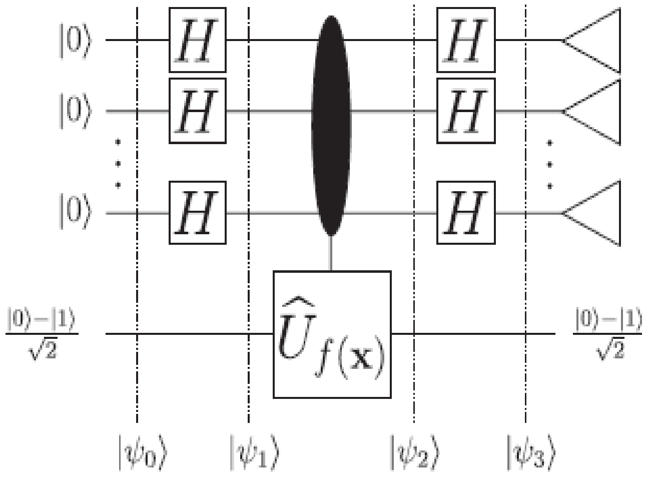


Figure IV-18: A circuit for the Deutsch–Jozsa algorithm. [37]

As we did for the Deutsch algorithm, we follow the state through the circuit. Initially the state is

$$\left|\psi_0\right\rangle = \left|0\right\rangle^{\otimes n}\left(\frac{\left|0\right\rangle - \left|1\right\rangle}{\sqrt{2}}\right) \quad \ldots \quad \textbf{\textit{(IV-42)}}$$

Consider the action of an n-qubit Hadamard transformation on the state $\left|0\right\rangle^{\otimes n}$:

$$H^{\otimes n}\left|0\right\rangle^{\otimes n} = \left(\frac{1}{\sqrt{2}}\right)^n \left(\left|0\right\rangle + \left|1\right\rangle\right)\otimes\left(\left|0\right\rangle + \left|1\right\rangle\right)\otimes\ldots\otimes\left(\left|0\right\rangle + \left|1\right\rangle\right) \quad \ldots \quad \textbf{\textit{(IV-43)}}$$

By expanding out the tensor product, this can be rewritten as

$$H^{\otimes n}\left|0\right\rangle^{\otimes n} = \frac{1}{\sqrt{2^n}}\sum_{x\in\{0,1\}^n}\left|x\right\rangle \quad \ldots \quad \textbf{\textit{(IV-44)}}$$

This is a very common and useful way of writing this state; the n-qubit Hadamard gate acting on the n-qubit state of all zeros gives a superposition of all $n$qubit basis states, all with the same amplitude $\frac{1}{\sqrt{2^n}}$ (called an ‘equally weighted superposition’). So the state immediately after the first $H^{\otimes n}$ in the Deutsch– Jozsa algorithm is

$$\left|\psi_1\right\rangle = \frac{1}{\sqrt{2^n}}\sum_{x\in\{0,1\}^n}\left|x\right\rangle\left(\frac{\left|0\right\rangle - \left|1\right\rangle}{\sqrt{2}}\right) \quad \ldots \quad \textbf{\textit{(IV-45)}}$$

Notice that the query register is now in an equally weighted superposition of all the possible n-bit input strings. Now consider the state immediately after the U_f (equivalently the c-$\hat{U}_{f(x)}$) gate. The state is

$$\left|\psi_2\right\rangle = \frac{1}{\sqrt{2^n}}U_f\,(\sum_{x\in\{0,1\}^n}\left|x\right\rangle\left(\frac{\left|0\right\rangle - \left|1\right\rangle}{\sqrt{2}}\right))$$

$$\left|\psi_2\right\rangle = \frac{1}{\sqrt{2^n}}\sum_{x\in\{0,1\}^n}(-1)^{f(x)}\left|x\right\rangle\left(\frac{\left|0\right\rangle - \left|1\right\rangle}{\sqrt{2}}\right) \quad \ldots \quad \textbf{\textit{(IV-46)}}$$

where we have associated the phase shift of $(-1)^{f(\mathbf{x})}$ with the first qubit.

To facilitate our analysis of the state after the interference is completed by the second Hadamard gate, consider the action of the n-qubit Hadamard gate on an n-qubit basis state $\left|x\right\rangle$.

It is easy to verify that the effect of the 1-qubit Hadamard gate on a 1-qubit basis state $|x\rangle$ can be written as

$$\begin{aligned} H|x\rangle &= \frac{1}{\sqrt{2}}\left(|0\rangle + (-1)^{x}|1\rangle\right) \\ &= \frac{1}{\sqrt{2}}\sum_{z\in\{0,1\}}(-1)^{xz}|z\rangle \end{aligned} \qquad \ldots \qquad \textbf{\textit{(IV-47)}}$$

Then we can see that the action of the Hadamard transformation on an *n*-qubit basis state $|x\rangle = |x_1\rangle|x_2\rangle..|x_n\rangle$ is given by

$$\begin{aligned} H^{\otimes n}|x\rangle &= H^{\otimes n}\left(|x_1\rangle|x_2\rangle..|x_n\rangle\right) = H|x_1\rangle H|x_2\rangle..H|x_n\rangle \\ &= \frac{1}{\sqrt{2}}\left(|0\rangle + (-1)^{x_1}|1\rangle\right)\frac{1}{\sqrt{2}}\left(|0\rangle + (-1)^{x_2}|1\rangle\right)..\frac{1}{\sqrt{2}}\left(|0\rangle + (-1)^{x_n}|1\rangle\right) \quad \ldots \quad \textbf{\textit{(IV-48)}} \\ &= \frac{1}{\sqrt{2^n}}\sum_{z_1,z_2,\ldots,z_n\in\{0,1\}^n}(-1)^{x_1z_1+x_2z_2+\ldots+x_nz_n}|z_1\rangle|z_2\rangle..|z_n\rangle \end{aligned}$$

The above equation above can be written more succinctly as

$$H^{\otimes n}|x\rangle = \frac{1}{\sqrt{2^n}}\sum_{z\in\{0,1\}^n}(-1)^{x.z}|z\rangle \qquad \ldots \qquad \textbf{\textit{(IV-49)}}$$

where **x** · **z** denotes the bitwise inner product of **x** and **z**, modulo 2 (we are able to reduce modulo 2 since $(-1)^2 = 1$). Note that addition modulo 2 is the same as the xor operation. The state after the final *n*-qubit Hadamard gate in the Deutsch–Jozsa algorithm is

$$\begin{aligned} |\psi_3\rangle &= \left(\frac{1}{\sqrt{2^n}}\sum_{x\in\{0,1\}^n}(-1)^{f(x)}\frac{1}{\sqrt{2^n}}\sum_{z\in\{0,1\}^n}(-1)^{x.z}|z\rangle\right)\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right) \\ &= \frac{1}{2^n}\sum_{z\in\{0,1\}^n}\left(\sum_{x\in\{0,1\}^n}(-1)^{f(x)+x.z}\right)|z\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right) \end{aligned} \qquad \ldots \qquad \textbf{\textit{(IV-50)}}$$

At the end of the algorithm a measurement of the first register is made in the computational basis (just as was done for the Deutsch algorithm). To see what happens, consider the total amplitude (coefficient) of $|z\rangle = |0\rangle^{\otimes n}$ in the first register of state $|\psi_3\rangle$. This amplitude is

$$\frac{1}{2^n}\sum_{x\in\{0,1\}^n}(-1)^{f(x)} \qquad \ldots \qquad \textbf{\textit{(IV-51)}}$$

Consider this amplitude in the two cases: *f* constant and *f* balanced. If *f* is constant, the amplitude of $|0\rangle^{\otimes n}$ is either +1 or −1 (depending on what value *f*(*x*) takes). So if *f* is constant, a measurement of the first register is *certain to return all 0s* (by ‘all 0s’ we mean the binary string 00 · · · 0). On the other hand, if *f* is balanced, then it is easy to see that the positive and negative contributions of the amplitudes cancel, and the overall amplitude of $|0\rangle^{\otimes n}$ is 0. So if *f* is balanced, a measurement of the first register *is certain not to return all 0s*. So to determine whether *f* is constant or balanced, the first register is measured. If the result of the measurement is all 0s, then the algorithm outputs ‘constant’, and otherwise it outputs ‘balanced’.

IV-6- Shor’s factoring algorithm [45]

IV-6-1- Exponential speed-up in Shor’s factoring algorithm

Shor’s algorithm is without doubt not only one of the cornerstones of quantum information theory but also one of the most surprising advances in the theory of computation itself: a problem, which is widely believed to be hard becomes tractable by referring to (quantum) physics – an approach completely atypical for the theory of computation, which usually abstracts away from any physical realization.

The problem Shor’s algorithm deals with is factorization, a typical NP problem. Consider for instance the task of finding the prime factors of 421301.With pencil and paper it might probably take more than an hour to find them. The inverse problem, the multiplication 601×701, can, however, be solved in a few seconds even without having pencil and paper at hand. The crucial difference between the two tasks multiplication and factoring is, however, how the degree of difficulty increases with he length of the numbers. Whereas multiplication belongs to the class of “tractable” problems for which the required number of elementary computing steps increases polynomially with the size of the input, every known classical factoring algorithm requires an exponentially increasing number of steps. This is what is meant by saying that factoring is an “intractable” or “hard” problem. In a nutshell the idea of Shor’s factoring algorithm is the following:

(1) Classical part: Using some elementary number theory one can show that the problem of finding a factor of a given integer is essentially equivalent to determining the period of a certain function. [34]

(2) Implement the function from step ***(1)*** in a quantum circuit and apply it to a superposition of all classical input states. Then perform a discrete quantum Fourier transform (QFT) and measure the output. The measurement outcomes will be probabilistically distributed according to the inverse of the sought period. The latter can thus be determined (with certain probability) by repeating the procedure.

(3) Efficient implementation: The crucial point of the algorithm is that the QFT as well as the function from step ***(1***) can be efficiently implemented, i.e., the number of required elementary operations grows only polynomially with the size of the input. Moreover, the probability of success of the algorithm can be made arbitrary close to one without exponentially increasing effort.

Clearly, the heart of the algorithm is an efficient implementation of the QFT. Since Fourier transforms enter in many mathematical and physical problems one might naively expect an exponential speedup for all these problems as well. However, the outcome of the QFT is not explicitly available but “hidden” in the amplitudes of the output state, which can not be measured efficiently. Only global properties of the function, like its period, can in some cases be determined efficiently.

Nevertheless, a couple of other applications are known for which the QFT leads again to an exponential speed up compared to the known classical algorithms. The abstract problem, which encompasses all these applications is known as the “hidden subgroup problem” and another rather prominent representative of this type is the discrete logarithm problem. Let us now have a more detailed look at the ingredients for Shor’s algorithm.

IV-6-2- Classical part

Let N be an odd number we would like to factor and a < N an integer which has no non-trivial factor in common with N, i.e., gcd(N, a) = 1. The latter can efficiently be checked by Euclid’s algorithm. A factor of N can then be found indirectly by determining the period p of the function $f : Z \rightarrow Z_N$ defined as $f(x) = a^x \text{ mod} N$.

Hence, we are looking for a solution of the equation $a^p - 1 = 0$ modN. Assuming p to be even we can decompose $a^p - 1 = (a^{p/2} + 1)(a^{p/2} - 1) = 0$ modN, and therefore either one or both terms $(a^{p/2} \pm 1)$ must have a factor in common with N. Any non-trivial common divisor of N with $(a^{p/2} \pm 1)$, again calculated by Euclid's algorithm, yields thus a non-trivial factor of N.

Obviously, the described procedure is only successful if p is even and the final factor is a non-trivial one. Fortunately, if we choose a at random, this case occurs with probability larger than one half unless N is a power of a prime. The latter can, however, be checked again efficiently by a known classical algorithm, which returns the value of the prime. Altogether a polynomial time algorithm for determining the period of the function above leads to a probabilistic polynomial time algorithm which either returns a factor of N or tells us that N is prime.

IV-6-3- Quantum Fourier Transform

The step from the ordinary discrete Fourier transform (based on matrix multiplication) to the Fast Fourier Transform (FFT) has been of significant importance for signal and image processing as well as for many other applications in scientific and engineering computing. Whereas the naive way of calculating the discrete Fourier transform

$$\hat{c}_y = \frac{1}{\sqrt{n}} \sum_{x=0}^{n-1} c_x e^{\frac{2\pi i}{n} xy} \qquad \ldots \qquad \textbf{\textit{(IV-52)}}$$

by matrix multiplication takes $O(n^2)$ steps, the FFT requires O(n log n). The quantum Fourier transform (QFT) is in fact a straightforward quantum generalization of the FFT, which can, however, be implemented using only $O((\log n)^2)$ elementary operations – an exponential speedup.

Let now the computational basis states of q qubits be characterized by the binary representation of numbers

$$|x\rangle = \sum_{i=1}^{q} x_i 2^{i-1} \text{ via } |x\rangle = |x_1,\ldots,x_q\rangle \qquad \ldots \qquad \textbf{\textit{(IV-53)}}$$

That is, in this subsection x denotes from now on a natural number or zero and not a binary word. Then for $n = 2^q$ the QFT acts on a general state vector of q qubits as

$$\sum_x c_x |x\rangle \rightarrow \sum_y c_y \hat{|} y\rangle \qquad \ldots \qquad \textbf{\textit{(IV-54)}}$$

This transformation can be implemented using only two types of gates: the Hadamard gate and conditional phase gates P$_d$ acting as $|a,b\rangle \rightarrow |a,b\rangle e^{\delta_{a+b,2}\pi_i \diagup_{2^d}}$, which rotate the relative phase conditionally by an angle $\pi 2^{-d}$, where d is the “distance” between the two involved qubits. ***Figure IV-19*** shows the quantum circuit, which implements the QFT on q = 3 qubits. The extension of the circuit to more than three qubits is rather obvious and since q(q + 1)/2 gates are required its complexity is O(q^2) = O((log n)^2). Being only interested in an approximate QFT we could reduce the number of gates even further to O(log n) by dropping all phase gates Pd with d ≥ m. Naturally, the accuracy will then depend on m.

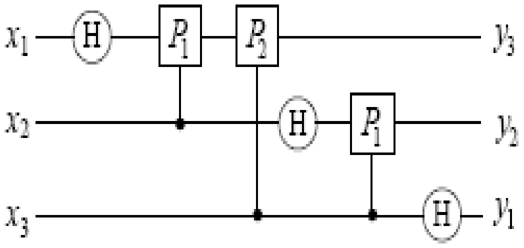


Figure V-19: The circuit of a discrete quantum Fourier transform on three qubits. [37]

IV-6-4- Joining the pieces

Let us now sketch how the QFT can be used to compute the period p of the function in the equation above efficiently. Consider two registers of q qubits each, where $2^q = n \geq N^2$ and all the qubits are in the state vector $|0\rangle$ initially. Applying a Hadamard gate to each qubit in the first register yields

$$\frac{1}{\sqrt{n}} \sum_x |x,0\rangle \qquad \ldots \qquad \textbf{\textit{(IV-55)}}$$

Now suppose we have implemented the function above in a quantum circuit which acts as $|x,0\rangle \rightarrow |x, f(x)\rangle$, where x is taken from Z$_n$. Applying this to the state vector and then performing a QFT on the first register we obtain

$$\frac{1}{n} \sum_{x,y=0}^{n-1} e^{\frac{2\pi i}{n} xy} |y, f(x)\rangle \qquad \ldots \qquad \textbf{\textit{(IV-56)}}$$

How will the distribution of measurement outcomes look like if we now measure the first register in computational basis? Roughly speaking, the sum over x will lead to constructive interference whenever y/n is close to a multiple of the inverse of the period p of f and it yields destructive interference otherwise. Hence, the probability distribution for measuring y is sharply peaked around multiples of n/p and p itself can be determined by repeating the whole procedure O(log N) times. At the same time the probability of success can be made arbitrary close to one. In the end we can anyhow easily verify whether the result, the obtained factor of N, is valid or not. What remains to be shown is that the map $|x,0\rangle \rightarrow |x, f(x)\rangle, \quad f(x) = a^x \bmod N$ can be implemented efficiently. This can be done by repeatedly squaring in order to get a^{2j} mod N and then multiplying a subset of these numbers according to the binary expansion of x. This requires O(log N) squarings and multiplications of log N-bit numbers. For each multiplication a standard algorithm requires $O((\log N)^2)$ steps. Hence, implementing this simple classical algorithm on our quantum computer we can compute f(x) with $O((\log N)^3)$ elementary operations. In fact, this part of performing a standard classical multiplication algorithm on a quantum computer is the bottleneck in the quantum part of Shor's algorithm. If there would be a more refined quantum modular exponentiation algorithm we could improve the asymptotic performance of the algorithm.

Altogether, the quantum part of Shor's factoring algorithm requires of the order $(\log N)^3$ elementary steps, i.e., the size of the circuit is cubic in the length of the input. As described above, additional classical preprocessing and postprocessing is necessary in order to obtain a factor of N. The time required for the classical part of the algorithm is, however, polynomial in logN as well, such that the entire algorithm does the job in polynomial time. In contrast to that, the running time of the number field sieve, which is currently the best classical factoring algorithm, is $\exp[O((\log N)^{1/3} (\log \log N)^{2/3})]$. Moreover, it is widely believed that factoring is a classically hard problem, in the sense that there exists no classical polynomial time algorithm. However, it is also believed that proving the latter conjecture (if it is true) is extremely hard since it would solve the notorious P = NP problem.

IV-7- Case of study “comparison between the linear ’classical’ search algorithm and Grover’s search algorithm”

As an illustration of how quantum algorithms are faster than classical algorithms, let us discuss the Grover’s search algorithm comparing to classical “linear” search algorithm.

IV-7-1- The classical search algorithm

In computer science, linear search is a search algorithm, also known as sequential search, which is suitable for searching a set of data for a particular value.

It operates by checking every element of a list one at a time in sequence until a match is found. Linear search runs in O(N). If the data are distributed randomly, on average (N+1)/2 comparisons will be needed. The best case is that the value is equal to the first element tested, in which case only 1 comparison is needed. The worst case is that the value is not in the list (or is the last item in the list), in which case N comparisons are needed.

The simplicity of the linear search means that if just a few elements are to be searched it is less trouble than more complex methods that require preparation such as sorting the list to be searched or more complex data structures, especially when entries may be subject to frequent revision. Another possibility is when certain values are much more likely to be searched for than others and it can be arranged that such values will be amongst the first considered in the list.

The following pseudocode describes the linear search technique.

```
For each item in the list:
  Check to see if the item you're looking for matches the item in the list.
    If it matches.
      Return the location where you found it.
    If it does not match.
      Continue searching until you reach the end of the list.
If we get here, we know the item does not exist in the list. Return -1.
```

In computer implementations, it is usual to search the list in order, from element 1 to N (or 0 to N - 1, if array indexing starts with zero instead of one) but a slight gain is possible by the reverse order. Suppose an array *A* having elements 1 to N is to be searched for a value *x* and if it is not found, the result is to be zero.

```
for i:=N:1:-1 do                %Search from N down to 1. (The step is -1)
 if A[i] = x then QuitLoop i;
next i;
Return(i);          %Or otherwise employ the value.
```

Implementations of the loop must compare the index value *i* to the final value to decide whether to continue or terminate the loop. If this final value is some variable such *N* then a subtraction *(i - N)* must be done each time, but in going down from *N* the loop termination condition is for a constant, and moreover a special constant. In this case, zero. Most computer hardware allows the sign to be tested, especially the sign of a value in a register, and so execution would be faster. In the case where the loop was for arrays indexed from zero, the loop would be *for i:=N - 1:0:-1 do* and the test on the index variable would be for it negative, not zero.

Finally this is the representation of the linear search algorithm by pascal programming language:

```
  Linear search algorithm
Function   linear {var A: list type: n, x : integer}: integer;
   Var
       i: integer;
       answer: integer;
   begin   {linear}
        answer:= 0;
         i:= 1
             while {i<= n} and {answer:=0} do
                 begin
                     if A{i}= x
                      then answer:=i
                      else incr {i}
                  end;
          linear:= answer
```

```
end;  {linear}
```

IV-7-2- Quantum Mechanical Algorithms

A good starting point to think of quantum mechanical algorithms is probabilistic algorithms. In these algorithms, instead of having the system in a specified state, it is in a distribution over various states with a certain probability of being in each state. At each step, there is a certain probability of making a transition from one state to another. The evolution of the system is obtained by premultiplying this probability vector (that describes the distribution of probabilities over various states) by a state transition matrix. Knowing the initial distribution and the state transition matrix, it is possible in principle to calculate the distribution at any instant in time.

Just like classical probabilistic algorithms, quantum mechanical algorithms work with a probability distribution over various states. However, unlike classical systems, the probability vector does not completely describe the system. In order to completely describe the system we need the *amplitude* in each state which is a complex number. The evolution of the system is obtained by premultiplying this amplitude vector (that describes the distribution of amplitudes over various states) by a transition matrix, the entries of which are complex in general. The probabilities in any state are given by the square of the absolute values of the amplitude in that state. It can be shown that in order to conserve probabilities, the state transition matrix has to be unitary.

The machinery of quantum mechanical algorithms is illustrated by discussing the three operations that are needed in Grover algorithm. The first is the creation of a configuration in which the amplitude of the system being in any of the 2^n basic states of the system is equal; the second is the Walsh-Hadamard transformation operation and the third the selective rotation of different states.

A basic operation in quantum computing is that of a "fair coin flip" performed on a single bit whose states are 0 and 1. This operation is represented by the following matrix:

$$M = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad \dots \qquad \textbf{\textit{(IV-57)}}$$

A bit in the state 0 is transformed into a superposition in the two states: $\left(\frac{1}{\sqrt{2}},\frac{1}{\sqrt{2}}\right)$. Similarly a bit in the state 1 is transformed into $\left(\frac{1}{\sqrt{2}},\frac{-1}{\sqrt{2}}\right)$ i.e. the magnitude of the amplitude in each state is $\frac{1}{\sqrt{2}}$ but the *phase* of the amplitude in the state 1 is inverted. The phase does not have an analog in classical probabilistic algorithms. It comes about in quantum mechanics since the amplitudes are in general complex. In a system in which the states are described by *n* bits (it has 2^n possible states) we can perform the transformation *M* on each bit independently in sequence thus changing the state of the system. The state transition matrix representing this operation will be of dimension 2^n X 2^n. In case the initial configuration was the configuration with all *n* bits in the first state, the resultant configuration will have identical amplitude of $2^{-\frac{n}{2}}$ in each of the 2^n states. This is a way of creating a distribution with the same amplitude in all 2^n states. Next consider the case when the starting state is another one of the 2^n states, i.e. a state described by an *n* bit binary string with some 0s and some 1s. The result of performing the transformation *M* on each bit will be a superposition of states described by all possible *n* bit binary strings with amplitude of each state having a magnitude equal to and sign either + or -. To deduce the sign, observe that from the definition of the matrix *M*, i.e. $M=\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}$, the phase of the resulting configuration is changed when a bit that was previously a 1 remains a 1 after the transformation is performed. Hence if $\bar{x}$ be the *n*-bit binary string describing the starting state and $\bar{y}$ the *n*-bit binary string describing the resulting string, the sign of the amplitude of $\bar{y}$ is determined by the parity of the bitwise dot product of $\bar{x}$ and $\bar{y}$, i.e. $(-1)^{\bar{x}.\bar{y}}$. This transformation is referred to as the Walsh-Hadamard transformation. This operation (or a closely related operation called the Fourier Transformation) is one of the things that makes quantum mechanical algorithms more powerful than classical algorithms and forms the basis for most significant quantum mechanical algorithms.

The third transformation that we will need is the selective rotation of the phase of the amplitude in certain states. The transformation describing this for a 4 state system is of the form:

$$\begin{bmatrix} e^{j\phi_1} & 0 & 0 & 0 \\ 0 & e^{j\phi_2} & 0 & 0 \\ 0 & 0 & e^{j\phi_3} & 0 \\ 0 & 0 & 0 & e^{j\phi_4} \end{bmatrix} \quad \ldots \qquad \textbf{\textit{(IV-58)}},$$

where $j = \sqrt{-1}$ and $\phi_1, \phi_2, \phi_3, \phi_4$ are arbitrary real numbers.

Note that, unlike the Walsh-Hadamard transformation and other state transition matrices, the probability in each state stays the same since the square of the absolute value of the amplitude in each state stays the same.

IV-7-3- The Problem

Let a system have $N = 2^n$ states which are labelled $S_1, S_2, \ldots S_N$. These 2^n states are represented as n bit strings. Let there be a unique state, say S_v, that satisfies the condition $C(S_v) = 1$, whereas for all other states S, $C(S) = 0$ (assume that for any state S, the condition $C(S)$ can be evaluated in unit time). The problem is to identify the state S_v.

IV-7-4- Algorithm

1- Initialize the system to the distribution: $\left(\frac{1}{\sqrt{N}}, \frac{1}{\sqrt{N}}, \ldots, \frac{1}{\sqrt{N}}\right)$, i.e. there is the same amplitude to be in each of the N states. This distribution can be obtained in *O(logN)* steps.

2- Repeat the following unitary operations $O\left(\sqrt{N}\right)$times

a- Let the system be in any state S:

In case C(S)=1 , rotate the phase by π radians;

In case C(S)=0 , leave the system unaltered.

b- Apply the diffusion transform *D* which is defined by the matrix *D* as follows:

$$D_{ij} = \frac{2}{N} \text{if } i \neq j \text{ and } D_{ii} = -I + \frac{2}{N} \quad \ldots \qquad \textbf{\textit{(IV-59)}}$$

This diffusion transform, *D*, can be implemented as $D=WRW$, where *R* the rotation matrix and *W* the Walsh Hadamard Transform Matrix are defined as follows:

R$_{ij}$=0 if $i \neq j$;

R$_{ii}$=1 if i=0; R$_{ii}$=-1 if $i \neq 0$.

As discussed in before:

$W_{ij} = 2^{-n/2}(-1)^{\bar{i}\,\bar{j}}$, where $\bar{i}$ is the binary representation of i , and $\bar{i}\,.\,\bar{j}$ denotes the bitwise dot product of the two *n* bit strings $\bar{i}$ and $\bar{j}$.

3- Sample the resulting state. In case *C(S$_v$)=1* there is a unique state *S*$_v$ such that the final state is *S*$_v$ with a probability of at least 1/2.

IV-7-5- Explanation of the algorithm

The loop in step *2*, is the heart of the algorithm. Each iteration of this loop increases the amplitude in the desired state by $O\left(\frac{1}{\sqrt{N}}\right)$, as a result in $O\sqrt{N}$ repetitions of the loop, the amplitude and hence the probability in the desired state reach *O(1)*. In order to see that the amplitude increases by $O\left(\frac{1}{\sqrt{N}}\right)$ in each repetition, we first show that the diffusion transform, *D*, can be interpreted as an *inversion about average* operation. A simple inversion is a phase rotation operation which is unitary.

In the following we show that the *inversion about average* operation is also a unitary operation and is equivalent to the diffusion transform *D* as used in step ***2-a*** of the algorithm.

Let α denote the average amplitude over all states, i.e. if α_i be the amplitude in the i^{th} state, then the average is $\frac{1}{N}\sum_{i=1}^{N}\alpha_i$. As a result of the operation *D*, the amplitude in each state increases (decreases) so that after this operation it is as much below (above) α as it was above (below) α before the operation.

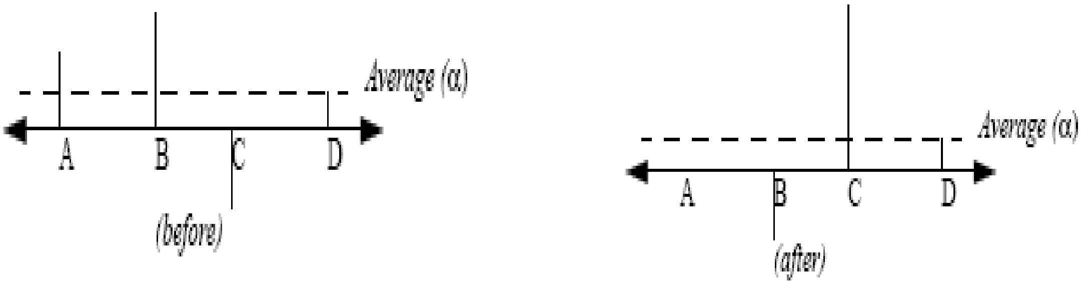


Figure IV-20: *Inversion about average* operation. [40]

The diffusion transform, *D*, is defined as follows:

$$D_{ij}=2/N \text{ if } i \neq j \text{ and } D_{ii}=-1+2/N \quad \ldots \qquad \textbf{(IV-60)}.$$

Next it is proved that *D* is indeed the *inversion about average* as shown in ***Figure IV-20*** . Observe that *D* can be represented in the form $D=-I+2P$ where *I* is the identity matrix and P is a projection matrix with $P_{ij}=1/N$ for all i,j. The following two properties of *P*are easily verified: first, that $P^2=P$ and second, that *P* acting on any vector gives a vector $\bar{v}$ each of whose components is equal to the average of all components.

Using the fact that $P^2=P$, it follows immediately from the representation $D=-I+2P$ that $D^2=I$ and hence *D* is unitary.

In order to see that *D* is the *inversion about average*, consider what happens when *D* acts on an arbitrary vector $\bar{v}$. Expressing *D* as $-I+2P$, it follows that: $D\bar{v}=(-I+2P)\bar{v}=-\bar{v}+2P\bar{v}$. Each component of the vector $\bar{v}$ is *A* where *A* is the average of all components of the vector $\bar{v}$. Therefore the i^{th} component of the vector is given by $(-v_i+2A)$ which can be written as $(A+(A-v_i))$ which is precisely the *inversion about average*.

Next consider what happens when the *inversion about average* operation is applied to a vector where each of the components, except one, are equal to a value, say *C*, which is approximately $\frac{1}{\sqrt{N}}$; the one component that is different is negative. The average *A* is approximately equal to *C*. Since each of the *(N-1)* components is approximately equal to the average, it does not change significantly as a result of the inversion about average. The one component that was negative to start out, now becomes positive and its magnitude increases by approximately *2C,* which is approximately $\frac{2}{\sqrt{N}}$.

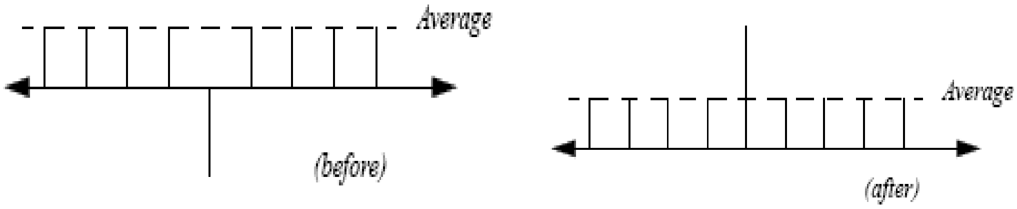


Figure IV-21: The *inversion about average* operation [40]

In the loop of step 2, first the amplitude in a selected state is inverted (this is a phase rotation and hence a valid quantum mechanical operation). Then the *inversion about average* operation is carried out. This increases the amplitude in the selected state in each iteration by $O\left(\frac{1}{\sqrt{N}}\right)$ (this is formally proved in ***theorem IV-1***).

Theorem IV-7 [40]

Let the state vector before step ***2-a*** *of the algorithm be as follows : for the one state that satisfies C(S)=1, the amplitude is k, for each of the (N-1) remaining states the amplitude is l such that* $\left(0 \prec k \prec \frac{1}{\sqrt{2}}\right) and\, l \succ 0$*. The change in k* (Δk) *after steps (a) and (b) of the algorithm is lower bounded by* $\Delta k \succ \frac{1}{2\sqrt{N}}$*. Also after steps (a) and (b),* $l \succ 0$ *.*

Using ***Theorem IV-7***, it immediately follows that there exists a number *M* less than $\sqrt{2N}$, such that in *M* repetitions of the loop in step 2, *k* will exceed $\frac{1}{\sqrt{2}}$. Since the probability of the system being found in any particular state is proportional to the square of the amplitude, it follows that the probability of the system being in the desired state when *k* is $\frac{1}{\sqrt{2}}$, is $k^2=1/2$. Therefore if the system is now sampled, it will be in the desired state with a probability greater than 1/2.

Theorem IV-8 [40]

D can be expressed as D=WRW, where W, the Walsh-Hadamard Transform Matrix and R, the rotation matrix, are defined as follows

$$\text{Rij=0 if } i \neq j \quad \ldots \tag{IV-61}$$

$$Rii=I \text{ if } i=0, Rii=-I \text{ if } i \neq 0 \quad \ldots \tag{IV-62}$$

$$W_{ij} = 2^{-n/2}(-1)^{\bar{i}\,\bar{j}} \quad \ldots \tag{IV-63}$$

Theorem IV-9 [40]

Let the state vector be as follows : for any one state the amplitude is k_1, for each of the remaining (N-1) states the amplitude is l_1. Then after applying the diffusion transform D, the amplitude in the one state is

$$k_2 = \left(\frac{2}{N}-1\right)k_1 + 2\frac{(N-1)}{N}l_1 \quad \ldots \tag{IV-64}$$

and the amplitude in each of the remaining (N-1) states is

$$l_2 = \frac{2}{N}k_1 + \frac{(N-2)}{N}l_1 \quad \ldots \tag{IV-65}$$

IV-7-6- How fast is it possible to find the desired element:

There is a matching lower bound that suggests that it is not possible to identify the desired element in fewer than $\Omega\left(\sqrt{N}\right)$steps. This result states that any quantum mechanical algorithm running for *T* steps is only sensitive to *$O(T^2)$*queries (i.e. if there are more possible queries, then the answer to at least one can be flipped without affecting the behaviour of the algorithm). So in order to correctly decide the answer which is sensitive to N queries will take a running time of $T=\Omega\left(\sqrt{N}\right)$. To see this assume that *C(S)=0* for all states and the algorithm returns the right result, i.e. that no state satisfies the desired condition. Then, if $T<\Omega\left(\sqrt{N}\right)$, the answer to at least one of the queries about C(S) for some *S* can be flipped without affecting the result, thus giving an incorrect result for the case in which the answer to the query was flipped.

IV-7-7- Implementation considerations

This algorithm is likely to be simpler to implement as compared to other quantum mechanical algorithms for the following reasons:

1- The only operations required are, first, the Walsh-Hadamard transform, and second, the conditional phase shift operation both of which are relatively easy as compared to operations required for other quantum mechanical algorithms.

2- Quantum mechanical algorithms based on the Walsh-Hadamard transform are likely to be much simpler to implement than those based on the "large scale Fourier transform".

3- The conditional phase shift would be much easier to implement if the algorithm was used in the mode where the function at each point was computed rather than retrieved form memory. This would eliminate the storage requirements in quantum memory.

4- In case the elements had to be retrieved from a table (instead of being computed), in principle it should be possible to store the data in classical memory and only the sampling system need be quantum mechanical. This is because only the system under consideration needs to undergo quantum mechanical interference, not the bits in the memory. What is needed, is a mechanism for the system to be able to *feel* the values at the various datapoints something like what happens in *interaction-free measurements*. Note that, in any variation, the algorithm must be arranged so as not to leave any trace of the path followed in the classical system or else the system would not undergo quantum mechanical interference.

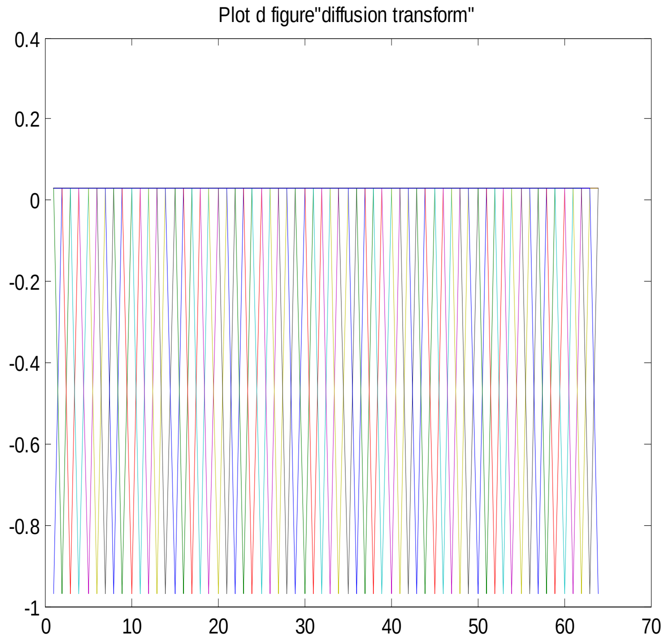
IV-7-8- The programme of Grover's search algorithm in MATLAB programme

```
%Search Alghorithm.
clear all;
%-----parameters-----------
nqubits=6;   %number of q-bits
n=2^nqubits;  %nnumber of elements in database
findmode=mod(round(n*rand+1),n);  %desired element
%-----defining quantum gates
d=-eye(n)+2/n;  %diffusion transform
```

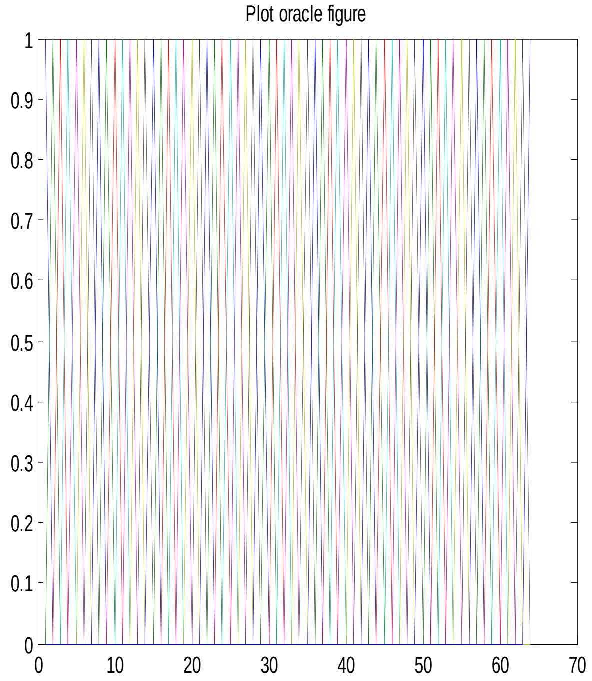
```
oracle=eye(n);  %oracle
oracle(findmode,findmode)=-1;
%--calculate the optimal number of iterations---
finish=round(pi/4*sqrt(n));
%--step(i)--initialization----
psistart=ones(n,1)/sqrt(n);
psi=psistart*exp(i*rand);
%step (ii)--algorithm body----
for steps=1:finish
steps
psi=d*oracle*psi;
probability(steps)=psi(findmode)*conj(psi(findmode));
end
%see the probability dynamics
plot(probability);
%see the result distribution
figure;
stem(psi.*conj(psi));
```

IV-7-9- Simulation of Grover algorithm with MATLAB programme: Example of 6nqubits :

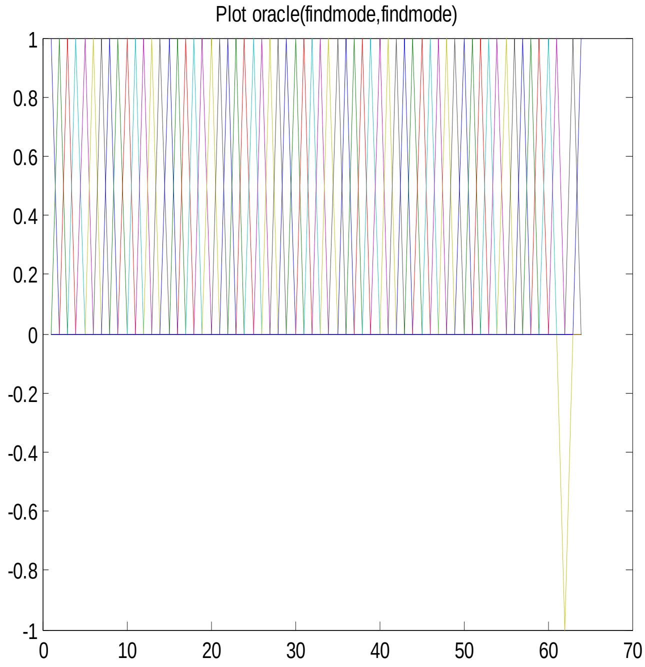
```
>> nqubits=6
nqubits = 6
>> n=2^nqubits
n =  64
>> findmode=mod(round(n*rand+1),n)
findmode = 62
>> d=-eye(n)+2/n;
>> plot(d)
```



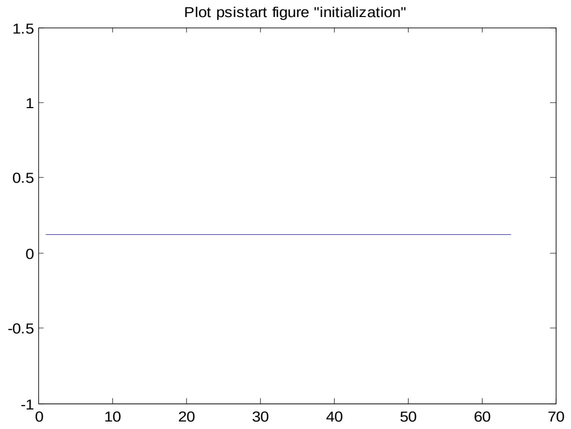
```
>> oracle=eye(n);

>> plot(oracle)
```



```
>> oracle(findmode,findmode)=-1;

>> plot(oracle)
```

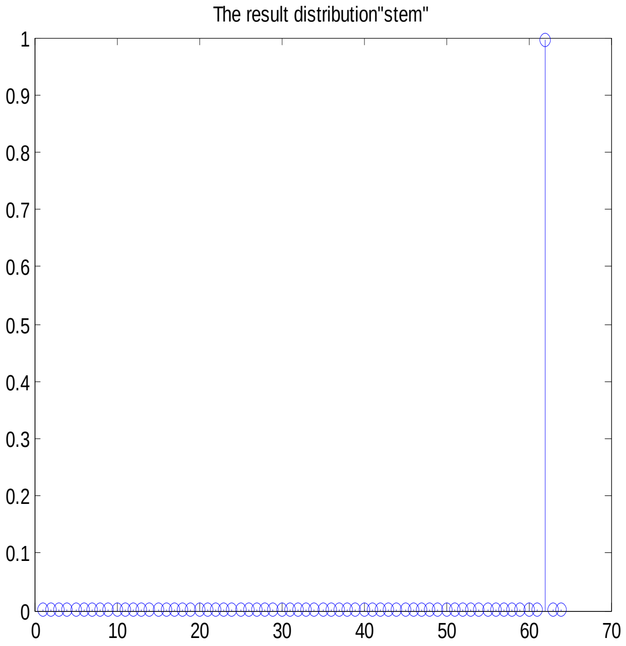


```
>> finish=round(pi/4*sqrt(n))

finish =  6

>> psistart=ones(n,1)/sqrt(n);

>> plot(psistart)
```



```
>> psi=psistart*exp(i*rand);

>> for steps=1:finish
```

```
Steps

psi=d*oracle*psi;

probability(steps)=psi(findmode)*conj(psi(findmode));

end

>> figure;

>> stem(psi.*conj(psi));
```



IV-8- Conclusion:

Quantum computers may solve some problems dramatically faster than conventional machines. One example is searching an unordered set for an item with specific properties. A quantum algorithm can find such an item (a "solution") in a time proportional to the square root of the size of the set, which is considerably faster than conventional ("classical") methods that take the same time as the size of the set.

Comparing to classical algorithm, with one solution out of 10 items, four steps of the quantum algorithm give less chance for a solution than two steps, showing the quantum algorithm can perform worse with more steps. And With one solution out of

1000 items, the quantum algorithm performs well with just 25 steps, which is much better than the classical method. Thus the control number of items and steps set the difficulty of the problem by changing the size of the set and the number of steps.

The classical algorithm is generate-and-test, that is, examine items one at a time until a solution is found. When the set has *n* items, the probability of finding a solution increases monotonically with the number of items examined until, after *n* steps; a solution is guaranteed to be found. The classical algorithm stops as soon as it finds a solution.

In the quantum algorithm, due to Lov Grover, the probability of finding a solution is close to 1 when the number of steps is about $\left(\pi/4\right)\sqrt{n}$. So the average number of steps in finding a solution is proportional to $\sqrt{n}$, much less than the linear growth with *n* for the classical algorithm. The quantum algorithm gives no answer until it completes the prespecified number of steps, and must restart from the beginning if it does not find a solution. Each repetition adds to the total number of steps required by the algorithm. Thus, if P_{steps} is the probability of finding a solution when run for a given number of steps, the average number of steps required to find a solution, including any repetitions is: steps/P_{steps}.

The probability for the quantum algorithm to find a solution oscillates with the number of steps. So taking more steps than needed to reach probability near 1 *decreases* the chance of finding a solution. Thus, the quantum algorithm requires care in selecting the number of steps. In addition, physical implementation of the quantum method in terms of qubits is simplest when the number of items is a power of two.

Finally, we can suspect that quantum computers work better than classical one because quantum computers need not limit themselves to checking each entry in succession. Instead, quantum computers can check several candidates at once using quantum parallelism. Unfortunately, the same quantum rules that let quantum computers feign parallelism exact a terrible price: they make it impossible to learn the individual outcomes of all the parallel computations, permitting instead only a collective property to be determined. Fortunately, this still offers enough of an advantage to let a quantum search beat a classical search.

During the past forty years astounding advances have been made in the manufacture of computers. The number of atoms needed to represent a bit in memory has been decreasing exponentially since 1950. Likewise the numbers of transistors per chip, clock speed, and energy dissipated per logical operation have all followed their own improving exponential trends. This rate of improvement cannot be sustained much longer; at the current rate in the year 2020 one bit of information will requite only one atom to represent it. The problem is that at that size the behaviour of a computer's components will be dominated by the principles of quantum physics.

As it is shown in the first chapter of our thesis when components shrink to where their behaviour will soon be dominated more by quantum physics than classical physics, researchers have begun to investigate the potential of these quantum behaviours for computation. These physical limitations of the classical computer and the possibility that the quantum computer can perform certain useful tasks more rapidly than any classical computer drive the study of quantum computing.

In chapter two after we present the important notions of quantum mechanics used in our thesis result, we move to talk about qubits. In a quantum computer, the fundamental unit of information (called a quantum bit or qubit), is not binary but rather more quaternary in nature. This qubit property arises as a direct consequence of its adherence to the laws of quantum mechanics which differ radically from the laws of classical physics. A qubit can exist not only in a state corresponding to the logical state 0 or 1 as in a classical bit, but also in states corresponding to a blend or superposition of these classical states.

In chapter three we discuss the progress which has been made in recent years in the experimental controlled manipulation of very small quantum systems that cannot be called other than spectacular, in a way that was not imaginable not long ago. Quantum gates have been implemented in the quantum optical context, and with nuclear magnetic resonance (NMR) techniques, single and coupled quantum dots experiment, charge and spin control in quantum dots experiment, spin relaxation and quantum dot quantum computing experiment using control over spins, even small quantum algorithms have been realized. Any such implementation will eventually have live up to some requirements that have may be most distinctly been formulated by DiVincenzo as generic requirements in practical quantum computation.

In parallel to the development of the application of quantum physics in the creation of quantum computers, also quantum algorithms were developed. In the fourth chapter we discuss the development of these algorithms which lead us to confirm the capacity and the speed of quantum computers as a final result of our work. In the same scientific paper in which David Deutsch introduced the notion of the universal quantum computer, he also presented the first quantum algorithm. Yet the Deutsch algorithm already exemplifies the advantages of a quantum computer through skillfully exploiting quantum parallelism. Like the Deutsch algorithm, other elementary quantum algorithms as Deutsch-Jozsa’s algorithm and Simon’s algorithm amount to deciding which black box out of finitely many alternatives one has at hand. Such a black box is often also referred to as oracle. An input may be given to the oracle, one may read out or use the outcome in later steps of the quantum algorithm, and the objective is to find out the functioning of the black box. It is assumed that this oracle operation can be implemented with some sequences of quantum logic gates.

Following Deutsch's algorithm, Shor demonstrated in 1994 that integers can be efficiently factorized on a quantum computer. It has lead to extensive work on developing new quantum algorithms. Finally we consider a case of study which is a comparison between Grover's algorithm or quantum "data base" search algorithm and classical "data base" search algorithm. As a result we find that Grover's algorithm allows a quantum computer to perform an unstructured search quadratically faster than any classical algorithm. This improves the capacity and the speed of quantum computers.

Now we know what purposes a quantum computer may serve, what tasks it may perform well, better than any classical computer, and have sketched what the underlying computational model is like. Also, ways have been described to fight decoherence that is due to the coupling to the environment, and eventually to the same devices that are designed to perform the read-out.

APPENDIX A

A-1- Lithography:

Optical lithography (photolithography) is a major application in the particle-matter interaction, and constitutes the classical process for fabricating integrated circuits. It is a key step in defining circuit patterns, and remains a barrier to any future development. Since resolution, at the out set, appears to be directly proportional to wave-length, feature size first progressed by a step-wise shortening of the wave length of the radiation used.

The operation works via a reduction lens system, by the exposure of photoresist film to energy particles from the Ultraviolet photons currently used through to x photons, ions, and finally electrons, all through a mask template carrying a pattern of the desired circuit. The aim of all this is to transfer this pattern on to a stack of insulating or conducting layers that make up the mask. These layers will have been deposited previously (the layering stage) on a wafer of semiconductor material, generally silicon. After this process, the resin dissolves under exposure to the air (development). The exposed parts of the initial layer can then be etched selectively, then the resin is lifted away chemically before deposition of the following layer. This lithography step can take place over twenty times during the fabrication of an integrated circuit.

In the 1980's, the micro electronics industry used mercury lamps delivering near UV through quartz optics, with an emission line of 436 nanometers (nm). This system was able to etch structures to a feature size of 3microns. This system was used through to the mid 90s, when it was replaced by excimer laser; which is a laser in which resonance cavity contains a halogen gas (for example, an argon fluorine mixture) and which delivers UV light pulses with durations in the nanosecond range and energies of the order of a few hundred mj; emitting far UV light (KrF, krypton fluoride at 248 nm then ArF, argon fluoride at 193 nm, with the photons thus created generating several electron volts) that were able to reach a resolution of 110 nm, pushed to under 90 nm with new processes.

In the 1980s, the CEA's (Electronics and Information Technology Laboratory) pioneered the application of lasers in lithography and the fabrication of integrated circuit production still uses these sources.

The next step for high volume production was expected to be the F2 laser (157 nm), but this lithography technology has to all intents and purposes been abandoned due to complications involved in producing optics in CaF2, which is transparent at this wave length. While the shortening of wave lengths in exposure tools has been the driving factor behind the strong resolution gain already achieved, two other factors have nevertheless played key roles. The first was the development of play mer-latice photoresist with low absorbance at the wave lengths used, implementing progressively more innovation input energy reflection/emission systems. The second was enhanced optics reducing diffraction interference (better surface quality, increase in numerical aperture).

Over the years, the increasing complexity of the optical systems has led to resolutions actually below the source wave-length. This development could not continue without a major technological break through a huge step forward in wave-length. For generations of integrated circuits with a lowest resolution of between 80 and 50 nm (the next node being at 65 nm), various different approaches are competing to offer particle projection at ever shorter wavelengths. They use either "soft" x-rays at extreme ultraviolet wavelength (around 10nm), "hard" x-ray at wavelengths below 1nm, ions or electrons.

The step crossing below the 50nm barrier will lead towards low-electron energy (10eV) enabled nano lithography with technology solution such as the scanning tunneling microscope and molecular beam epitaxy for producing "superlattices".

A-2- Molecular beam epitaxy:

Quantum wells are grown using Molecular Beam Epitaxy (from the greek taxi, meaning order, and epi, meaning over), or MBE. The principle of this physical deposition technique, which was first developed for growing III-V semiconductor crystals, is based on the evaporation of ultra-pure elements of the component to be grown, in a furnace under ultra-high vacuum (where the pressure can be as low as 5.10^{-11} mbar) in order to create a pure, pollution-free space.

One or more thermal beams of atoms or molecules react on the surface of a single-crystal wafer placed on a substrate kept at high temperature (several hundred °C), which serves as a lattice for the formation of a film called epitaxial film. It thus becomes possible to stack ultra-thin layers that measure a millionth of millimeter each; ie: composed of only a few atom planes.

The elements are evaporated or sublimated from an ultra-pure source placed in effusion cell (or Knudsen cell; an enclosure where a molecular flux moves from a region with a given pressure to another region of lower pressure) heated by the Joule effect.

A range of structural and analytical probes can monitor film growth in situ in real time, particularly using surface quality analysis and grazing angle phase transitions by LEED (Low Energy Electron Diffraction) or RHEED (Reflection High Energy Electron Diffraction). Various spectroscopic methods are also used, including Auger electron spectroscopy, Secondary Ion Mass Spectrometry (SIMS), X-ray Photoelectron Spectrometry (XPS) or Ultraviolet Photon Electron Spectrometry (UPS).

As ultra-high vacuum technology has progressed, molecular beam epitaxy has branched out to be applied beyond III-V semiconductors to embrace metals and insulators. In fact, the vacuum in the growth chamber, whose design changes depending on the properties of the matter intended to be deposited, has to be better than 10^{-11}mbar in order to grow an ultra-pure film of exceptional crystal quality at relatively low substrate temperatures. This value corresponds to the vacuum quality when the growth chamber is a trest. Arsenide's, for example, grow at a residual vacuum of around 10^{-8}mbar as soon as the arsenic cell has reached its set growth the temperature.

The pumping necessary to achieve these performance levels draws on several techniques using ion pumps, cryopumping, titanium sublimation pumping, diffusion pumps or turbo-molecular pumps. The main impurities (H_2, H_2O, CO, and CO_2) can present partial pressures of lower than 10^{-13}mbar.

A-3- Phase Kick-Back

When described in the classical basis, the CNOT gate appears to do nothing to the control qubit, it can in fact affect the control qubit just as much as it does the target qubit. For example, in the Hadamard basis, the role of control and target qubit is effectively switched, for example,

$$CNOT:\left(\frac{|0\rangle+|1\rangle}{\sqrt{2}}\right)\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)\mapsto\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)$$

Notice that $\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)$is an eigenvector (or *eigenstate*) of the *X*(NOT) gate with eigenvalue −1, and an eigenvector of the identity gate with eigenvalue +1. Since the CNOT applies the NOT gate to the target qubit if the first qubit is in state$|1\rangle$, we get

$$CNOT:|1\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)\mapsto|1\rangle\left(NOT\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)\right)=|1\rangle\left((-1)\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)\right)=-|1\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)$$

Since the CNOT applies the identity gate (i.e. does 'nothing') to the target qubit if the first qubit is in state$|0\rangle$, we get

$$CNOT:|0\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)\mapsto|0\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)$$

Since the target qubit is in an eigenstate, it does not change, and we can effectively treat the eigenvalue as being 'kicked back' to the control register.

Note that this can be summarized as

$$CNOT:|b\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)\mapsto(-1)^b|b\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)$$

where$b\in\{0,1\}$. When the control qubit is in a superposition of $|0\rangle$ and $|1\rangle$, we have

$$CNOT:(\alpha_0|0\rangle+\alpha_1|1\rangle)\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)\mapsto(\alpha_0|0\rangle-\alpha_1|1\rangle)\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)$$

(notice this corresponds to effecting the *Z* gate to the control qubit).

Let us consider the effect of a more general 2-qubit gate *Uf* implementing an arbitrary function $f:\{0,1\}\mapsto\{0,1\}$by mapping $U_f=|x\rangle|y\rangle\mapsto|x\rangle|y\oplus f(x)\rangle$(this mapping is reversible even though the function *f* may not itself be invertible).

Let us fix the target register to the state$\frac{1}{\sqrt{2}}(|0\rangle-|1\rangle)$, and analyse the action of U_f on an arbitrary basis state in the control qubit:

$$U_f : |x\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)\mapsto\left(\frac{U_f|x\rangle|0\rangle-U_f|x\rangle|1\rangle}{\sqrt{2}}\right)=\frac{|x\rangle|0\oplus f(x)\rangle-|x\rangle|1\oplus f(x)\rangle}{\sqrt{2}}$$
$$=|x\rangle\left(\frac{|0\oplus f(x)\rangle-|1\oplus f(x)\rangle}{\sqrt{2}}\right)$$

We know that the action of '$\oplus f(x)$' has no effect on a single bit if $f(x)=0$ (i.e. $b\oplus 0=b$), and '$\oplus f(x)$' flips the state of the bit if $f(x)=1$.

Consider the expression $\frac{1}{\sqrt{2}}\left(|0\oplus f(x)\rangle-|1\oplus f(x)\rangle\right)$ in the two cases $f(x)=0$ and $f(x)=1$:

$$f(x)=0:\frac{1}{\sqrt{2}}\left(|0\oplus f(x)\rangle-|1\oplus f(x)\rangle\right)=\frac{|0\rangle-|1\rangle}{\sqrt{2}}$$
$$f(x)=1:\frac{1}{\sqrt{2}}\left(|0\oplus f(x)\rangle-|1\oplus f(x)\rangle\right)=\frac{|1\rangle-|0\rangle}{\sqrt{2}}=-\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)$$

These two possibilities differ by a factor of (-1) which depends on the value of $f(x)$. We have

$$\frac{1}{\sqrt{2}}\left(|0\oplus f(x)\rangle-|1\oplus f(x)\rangle\right)=(-1)^{f(x)}\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)$$

So the above state can be rewritten as

$$|x\rangle(-1)^{f(x)}\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)$$

Associating the $(-1)^{f(x)}$ factor with the first qubit, we have

$$U_f : |x\rangle\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)\mapsto|x\rangle(-1)^{f(x)}\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)$$

When the control qubit is in a superposition of $|0\rangle$ and $|1\rangle$, we have

$$U_f : \left(\alpha_0|0\rangle+\alpha_1|1\rangle\right)\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)\mapsto\left((-1)^{f(0)}\alpha_0|0\rangle+(-1)^{f(1)}\alpha_1|1\rangle\right)\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)$$

We can think of U_f as a 1-qubit operator .$\hat{U}_{f(x)}$ (which maps $|b\rangle\mapsto|b\oplus f(x)\rangle$) acting on the second qubit, *controlled* by the state $|x\rangle$ of the first register, as shown in Figure 6.6.

We may sometimes write $c-\hat{U}_{f(x)}$ instead of U_f.

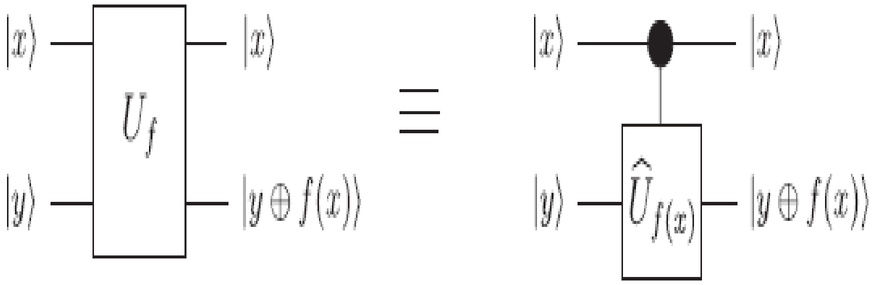


Figure A-1: The 2-qubit gate $U_f : |x\rangle|y\rangle \mapsto |x\rangle|y \oplus f(x)\rangle$ can be thought of as a 1-qubit gate .$\hat{U}_{f(x)}$ acting on the second qubit, controlled by the first qubit.

Notice in that the state $\frac{|0\rangle - |1\rangle}{\sqrt{2}}$ of the second register is an eigenvector of $\hat{U}_{f(x)}$.

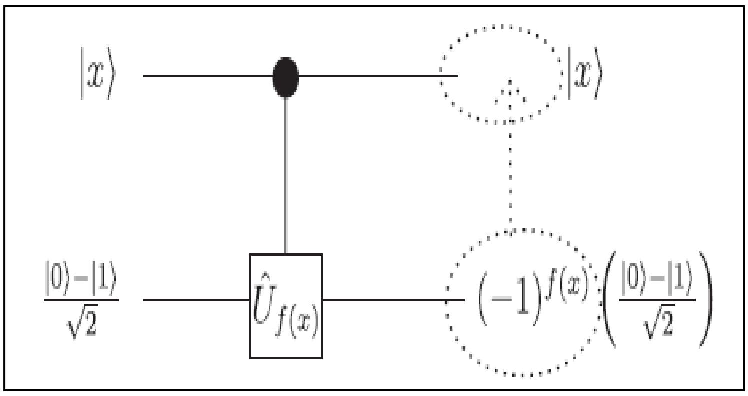

$|x\rangle$
$|x\rangle$
$\frac{|0\rangle-|1\rangle}{\sqrt{2}}$
$\hat{U}_{f(x)}$
$(-1)^{f(x)}\left(\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right)$

Figure A-2: The state $\frac{|0\rangle - |1\rangle}{\sqrt{2}}$ of the target register is an eigenstate of $\hat{U}_{f(x)}$. The eigenvalue $(-1)^{f(x)}$ can be ‘kicked back’ in front of the target register.

APPENDIX B

B-1- Fourier Transforms

The Fourier transform defines a relationship between a signal in the time domain and its representation in the frequency domain. Being a transform, no information is created or lost in the process, so the original signal can be recovered from knowing the Fourier transform, and vice versa.

The Fourier transform itself is defined by the equation

$$X(f) = \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}dt$$

where *X(f)* is the Fourier transform of *x(t)* Frequency is measured in Hertz, with *f*as the frequency variable.

B-2- Fast Fourier Transform

The fast Fourier transform (FFT) is a discrete Fourier transform algorithm which reduces the number of computations needed for *N* points from $2N^2$to *2N lg N*, where lg is the base-2 logarithm. If the function to be transformed is not harmonically related to the sampling frequency, the response of an FFT looks like a sinc function (although the integrated power is still correct). Aliasing (leakage) can be reduced by apodization using a tapering function. However, aliasing reduction is at the expense of broadening the spectral response.

Fast Fourier transform algorithms generally fall into two classes: decimation in time, and decimation in frequency. The Cooley-Tukey FFT algorithm first rearranges the input elements in bit-reversed order, then builds the output transform (decimation in time). The basic idea is to break up a transform of length *N* into two transforms of length *N/2* using the identity

VI- The Quantum Fourier Transform

Let us assume that we start with the state $|x\rangle = |x_{N-1}\rangle |x_{N-2}\rangle .. |x_1\rangle |x_0\rangle$

which is the bit representation of the *N* digit number *x* with x_0 being the value of the least significant bit, and x_{n-1} that value of the most significant bit. Thus the number *x* is given by $x = \sum_{m=0}^{N-1} x_m 2^m$ with the x_m taking the values 0 or 1. We now want to take this state to the state $F\left|x\right\rangle = \frac{1}{2^{N/2}} \sum_k e^{i2\pi kx2^{-N}} \left|k_0 ... k_N\right\rangle$ where *k* is the number represented by

$$k = \sum_{n=0}^{N-1} k_n 2^n$$

Note that we have reversed the representation for *k* so that the least significant bit is represented by the first qubit state, and the most significant is represented by the last qubit state.

Now, the phase factor can be rewritten as

$$e^{i2\pi kx2^{-N}} = e^{i2\pi} \sum\nolimits_{n=0}^{N-1} k_n \sum\nolimits_{m=0}^{N-1} x_m 2^{n+m-N} = e^{i2\pi} \sum\nolimits_{n=0}^{N-1} k_n \sum\nolimits_{m=0}^{N-1-n} x_m 2^{n+m-N}$$

since $e^{i2\pi 2^{n+m-N}} = 1$ if *0<n+m-N*. We thus notice that the phase for any given value of *n* (ie the *n-th* least significant bit of *k*) depends only on the values of the bits of *x* of order less that *N-1-n*. If we line up the bit representations of *k* and *x* we have

x_{N-1}	x_{N-2}	…	x_{N-1-n}	…	x_1	x_0
k_0	k_1	…	k_n	…	k_{N-2}	k_{N-1}

The Fourier factor which depends on k_n is

$$e^{i2\pi k_n} \sum\nolimits_{m=0}^{N-1-n} x_m 2^{n+m-N} = e^{i2\pi k_n} \sum\nolimits_{m=0}^{N-1-n} x_{N-1-n-m} 2^{-(m+1)}$$

and depends only on those bits of the representation of *x* which lie at or to the right of that bit in the representation of *k*. Furthermore, we note that in the factor which depends on k_n, the phase which depends on the largest *x* bit, namely x_{N-1-n} is $e^{i\pi k_n x_{N-1-n}}$ which has only values of plus or minus 1.

We can now perform the Fourier Transform bit by bit starting with the lowest digit of *k*, namely k_0. Let me assume that we have managed to transform the state $\left|x\right\rangle$ by replacing the *r-1* highest digits of *x* with the lowest *r-1* digits of *k*. *I.e.*, I have created, by some sequence of transformations, the state

$$|x\rangle \rightarrow \frac{1}{2^{\frac{(r-1)}{2}}} \sum_{\{k_0 \ldots k_{r-1}\}=\{0,1\}} e^{i2\pi \sum_{n=0}^{r-1} k_n \sum_{m=0}^{N-r} x_{N-r-m} 2^{-(m+1)}} |k_0 k_1 .. k_{r-1} x_{N-1-r} .. x_0\rangle$$

We now show how to advance this to next stage where we will create the expression up to the *rth* bit. We can accomplish this by generating a transformation of the form

$$\sum |k_0 k_1 .. k_{r-1} x_{N-1-r} .. x_0\rangle$$

$$\frac{1}{\sqrt{2}} |k_0 .. k_{r-1}\rangle \left(\sum_{k_r}^{1} \left[e^{i\pi k_r x_{N-1-r}} |k_r\rangle \right] e^{i\pi k_r} \sum_{m=1}^{N-1-r} x_{N-1-r-m} 2^{-m} \right) |x_{N-r-2} .. x_0\rangle$$

This transformation can be decomposed into the two sets of transformations

$$|k_0 .. k_{r-1}\rangle |x_{N-1-r}\rangle |x_{N-r-2} .. x_0\rangle$$

$$e^{i2\pi k_r} \sum_{m=1}^{N-1-r} x_{N-1-r-m} 2^{-m} |k_0 .. k_r\rangle |x_{N-r-2} .. x_0\rangle$$

and

$$|k_0 .. k_r\rangle |x_{N-r-2} .. x_0\rangle$$

$$e^{i2\pi k_r} \sum_{m=1}^{N-1-r} x_{N-1-r-m} 2^{-m} |k_0 .. k_r\rangle |x_{N-r-2} .. x_0\rangle$$

The first transformation is just a $\pi/2$ rotation of the *rth* bit.

$$|0\rangle \rightarrow \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle)$$

$$|1\rangle \rightarrow \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle)$$

The second set just corresponds to a series of controlled one bit phase rotations.

$$e^{i2\pi k_r} \sum_{m=1}^{N-1-r} x_{N-1-r-m} 2^{-m} |k_0 \ldots k_r\rangle |x_{N-r-2} \ldots x_0\rangle = |k_0 \ldots k_r\rangle \prod_{m=1}^{N-1-r} \left(e^{2\pi k_r x_{N-1-m-r} 2^{-(m+1)}} |x_{N-1-r-m}\rangle \right)$$

I.e., these are transformations which phase rotate the $|x_{N-1-r-m}\rangle$ bit depending on whether $|k_r\rangle$ bit is one or zero.

Thus given the transform up to *r-1* bits, it requires a single $\pi/2$ rotation of a single bit, and *N-r-1* controlled single-bit phase rotations, for a total of *N-r* operations. Thus the whole Fourier transformation requires $\sum_{r=0}^{N-1} (N-r) = N(N+1)/2$ operations (*N*, we recall is the number of bits in each of the numbers).

If we apply this Fourier transform to a state of the form $\left|\phi\right\rangle = \sum_x \alpha_x \left|x\right\rangle$ we get for the Quantum Fourier Transform

$$QFT\left|\phi\right\rangle = \sum_x \alpha_x \sum_k \frac{1}{2^{N/2}} e^{i2\pi kx}\left|k\right\rangle = \sum_k \left(\sum_x \frac{1}{2^{\frac{N}{2}}} e^{i\pi kx}\alpha_x \right)\left|k\right\rangle$$

The term in the brackets is just the discrete Fourier transform of α_x.

Note again that the representation $\left|k\right\rangle$ is the bit reversed of the representation of *x*. While one could do a bit reversal operation to get $\left|k\right\rangle$ into the same bit order as $\left|x\right\rangle$, there is no point.

$$\sum_{n=0}^{N-1} a_n e^{-2\pi i n k/N} = \sum_{n=0}^{N/2-1} a_{2n} e^{-2xi(2n)k/N} + \sum_{k=0}^{N/2-1} a_{2n+1} e^{-2\pi i(2x+1)k/N}$$

$$\sum_{n=0}^{N-1} a_n e^{-2\pi i n k/N} = \sum_{n=0}^{N/2-1} a_n e^{-2\pi i n k(N/2)} + e^{-2\pi n k/N} \sum_{k=0}^{N/2-1} a_n e^{-2\pi i n k\left(N/2\right)}$$

APPENDIX C

C-1- Grover MATLAB program commands definition and description

a- Round: Round to nearest integer

Syntax: Y = round(X)

Description: Y = round(X) rounds the elements of X to the nearest integers. For complex X, the imaginary and real parts are rounded independently.

b- rand: Uniformly distributed pseudorandom numbers

Syntax:

Y = rand

Y = rand(n)

Y = rand(m,n)

Y = rand([m n])

Y = rand(m,n,p,...)

Y = rand([m n p...])

Y = rand(size(A))

rand(method,s)

s = rand(method)

Description: Y = rand returns a pseudorandom, scalar value drawn from a uniform distribution on the unit interval.

Y = rand(n) returns an n-by-n matrix of values derived as described above.

Y = rand(m,n) or Y = rand([m n]) returns an m-by-n matrix of the same.

Y = rand(m,n,p,...) or Y = rand([m n p...]) generates an m-by-n-by-p-by-... array of the same.

Y = rand(size(A)) returns an array that is the same size as A.

rand(method,s) causes rand to use the generator determined by method, and initializes the state of that generator using the value of s.

The value of s is dependent upon which method is selected. If method is set to 'state' or 'twister', then s must be either a scalar integer value from 0 to 2^32-1 or the output of rand(method). If method is set to 'seed', then s must be either a scalar integer value from 0 to 2^31-2 or the output of rand(method).

c- mod : Modulus after division

Syntax : M = mod(X,Y)

Description: M = mod(X,Y) if Y ~= 0, returns X - n.*Y where n = floor(X./Y). If Y is not an integer and the quotient X./Y is within roundoff error of an integer, then n is that integer. The inputs X and Y must be real arrays of the same size, or real scalars.

The following are true by convention: mod(X,0) is X mod(X,X) is 0 mod(X,Y) for X~=Y and Y~=0 has the same sign as Y.

Examples:

mod(13,5)

ans = 3

d- eye : Identity matrix

Syntax :

Y = eye(n)

Y = eye(m,n)

Y = eye(size(A))

eye(m, n, classname)

eye([m,n],classname)

Description:

Y = eye(n) returns the n-by-n identity matrix.

Y = eye(m,n) or eye([m n]) returns an m-by-n matrix with 1's on the diagonal and 0's elsewhere.

Y = eye(size(A)) returns an identity matrix the same size as A.

eye(m, n, classname) or eye([m,n],classname) is an m-by-n matrix with 1's of class classname on the diagonal and zeros of class classname elsewhere. classname is a string specifying the data type of the output. classname can have the following values: 'double', 'single', 'int8', 'uint8', 'int16', 'uint16', 'int32', 'uint32', 'int64', or 'uint64'.

Example: x = eye(2,3,'int8');

Limitations: The identity matrix is not defined for higher-dimensional arrays. The assignment y = eye([2,3,4]) results in an error.

e- ones: Create an array of all ones

Syntax :

Y = ones(n)

Y = ones(m,n)

Y = ones([m n])

Y = ones(d1,d2,d3...)

Y = ones([d1 d2 d3...])

Y = ones(size(A))

ones(m, n,...,classname)

ones([m,n,...],classname)

Description:

Y = ones(n) returns an n-by-n matrix of 1s. An error message appears if n is not a scalar.

Y = ones(m,n) or Y = ones([m n]) returns an m-by-n matrix of ones.

Y = ones(d1,d2,d3...) or Y = ones([d1 d2 d3...]) returns an array of 1s with dimensions d1-by-d2-by-d3-by-....

Y = ones(size(A)) returns an array of 1s that is the same size as A.

ones(m, n,...,classname) or ones([m,n,...],classname) is an m-by-n-by-... array of ones of data type classname. classname is a string specifying the data type of the output. classname can have the following values: 'double', 'single', 'int8', 'uint8', 'int16', 'uint16', 'int32', 'uint32', 'int64', or 'uint64'.

Example: x = ones(2,3,'int8');

f- plot: Plot data series

Syntax:

plot(tsobj)

hp = plot(tsobj)

plot(tsobj, linefmt)

hp = plot(tsobj, linefmt)

plot(..., volumename, bar)

hp = plot(..., volumename, bar)

Description:

plot(tsobj) plots the data series contained in the object tsobj. Each data series will be a line. plot automatically generates a legend as well as dates on the x-axis. Grid is turned on by default. plot uses the default color order as if plotting a matrix.

The plot command automatically creates subplots when multiple time series are encountered, and they differ greatly on their decimal scales. For example, subplots are generated if one time series data set is in the 10s and another's is in the 10,000s.

hp = plot(tsobj) additionally returns the handle(s) to the object(s) inside the plot figure. If there are multiple lines in the plot, hp is a vector of multiple handles.

plot(tsobj, linefmt) plots the data series in tsobj using the line format specified. For a list of possible line formats, see plot in the MATLAB documentation. The plot legend is not generated, but the dates on the x-axis and the plot grid are. The specified line format is applied to all data series; that is, all data series will have the same line type.

hp = plot(tsobj, linefmt) plots the data series in tsobj using the format specified. The plot legend is not generated, but the dates on the x-axis and the plot grid are. The specified line format is applied to all data series, that is, all data series can have the same line type. If there are multiple lines in the plot, hp is a vector of multiple handles.

plot(..., volumename, bar) additionally specifies which data series is the volume. The volume is plotted in a subplot below the other data series. If bar = 1, the volume is plotted as a bar chart. Otherwise, a line plot is used.

hp = plot(..., volumename, bar) returns handles for each line. If bar = 1, the handle to the patch for the bars is also returned.

g- Stem: Two-Dimensional Stem Plots

A stem plot displays data as lines (stems) terminated with a marker symbol at each data value. In a 2-D graph, stems extend from the x-axis.

The stem function displays two-dimensional discrete sequence data. For example, evaluating the function with the values

alpha = .02; beta = .5; t = 0:4:200;

y = exp(-alpha*t).*cos(beta*t);

yields a vector of discrete values for y at given values of t. A line plot shows the data points connected with a straight line.

plot(t,y)

A stem plot of the same function plots only discrete points on the curve.

stem(t,y)

Add axes labels to the x- and y-axis.

[1] http://public.irts.net, The 2003 International Technology Roadmap for Semiconductors

[2] Shooman M., *Reliability of Computer System and Networks: Fault Tolerance,* Analysis. and Design, Wiley Interscience Publication, New York 2002.

[3] Flynn M., et al., *Deep-Submicron Microprocessor Design Isues*, IEEE Micro, Vol. 19, No. 4, 1999, pp. 11-22

[4] Plummer J., Griffin P., *Material and Process Limits in Silicon VLSI Technology*, Proceeedings of the IEEE, vol. 89, No. 3, March 2001, pp. 240-258.

[5] www.Wikipedia.com/transistors

[6] The national technology Roadmap for semiconductors, tech. report, Semiconductor Industry Assn., San Jose, Calif., 1994 and 1997 (updated).

[7] K.Nowka, High performance CMOS System Design Using Wave Pipelining, Phd thesis, Stanford Univ., Dept., Electrical Eng ., 1995.

[8] Varadarajan at al., *Low Power Design Issues*, in The Computer Engineering Handbook, V. Oklobdzija, Ed, CRC Press, Baca Raton, 2002..

[9] http://www.sematech.org/, Semiconductor Industry Association (03) The National Technology Roadmap for Semiconductors (online)

[10] Patt Y., *Requirements, Bottlenecks, and Good Fortune: Agents for Microprocessor Evolution*, Proc. of the IEEE, Vol., 89, No. 11 2001, pp.1553- 1559.

[11] Burger D., Goodman J., *Billion-Transistor Architectures: There and Back Again*, IEEE Computer, Vol. 37, No. 3, 2004, pp. 22-28.

[12] Mile Stojˇcev, Teufik Toki´c and Ivan Milentijevi´, *The limits of semiconductor technology and oncoming challenges in computer microarchitectures and architectures*.

[13] www.Wikipedia.com/Sterm-Gerlach experiment

[14] N. Mott, Proc. Roy. Soc. A 153, 699 (1936).

[15] Stefano Sanvito ‘Ab-initio methods for spin-transport at the nanoscale level’

[16] G. Binasch, P. Gr¨unberg, F. Saurenbach, and W. Zinn, Phys. Rev. B 39, 4828 (1989).

[17] S. Parkin, N. More, and K. Roche, Phys. Rev. Lett. 64, 2304 (1990).

[18] K. Tsukagoshi, B. W. Alphenaar, and H. Ago, Nature 401, 572 (1999).

[19] B. Zhao, I. M¨onch, T. M¨uhl, H. Vinzelberg, and C. Schneider, J. Appl. Phys. 91, 7026 (2002).

[20] B. Zhao, I. M¨onch, H. Vinzelberg, T. M¨uhl, and C. Schneider, Appl. Phys. Lett. 80, 3144 (2002).

[21] S. Sahoo, T. Kontos, C. Sch¨onenberger, and C. S¨urgers, cond-mat/0411623 .

[22] A. Jensen, J. Nygard, and J. Borggreen, *in Toward the controllable quantum states,* Proceedings of the International Symposium on Mesoscopic Superconductivity and Spintronics, H. Takayanagi and J. Nitta , 33 (2003).

[23] Pkilip Kaye, *Raymond Laflamme and Michele Mosca, An introduction to Quantum computing,* Oxfor University Publisher.

[24] Dirk Bouwmeester, Artur Ekert, Anton Zeilinger (Eds.), *The Physics of Quantum Information, Quantum Cryptography,Quantum Teleportation, Quantum Computation*, Springer (2000).

[25] Schleser R, Ruh E, Ihn T, Ensslin K, Driscoll D C and Gossard A C 2004 Appl. Phys. Lett. 85 2005

[26] Field M, Smith C G, Pepper M, Ritchie D A, Frost J E F, Jones G A C and Hasko D G 1993 Phys. Rev. Lett. 70 1311

[27] Petta J R, Johnson A C, Marcus C M, Hanson M P and Gossard A C 2004 Phys. Rev. Lett. 93 186802

[28] Kodera T, van der Wiel W G, Ono K, Sasaki S, Fujisawa T and Tarucha S 2004 Physica E 22
518, 2004

[29] Lee H, Johnson J A, Speck J S and Petroff P M 2000 J. Vac. Sci. Technol. B 18 2193

[30] Engel H-A, Kouwenhoven L P, Loss D and Marcus C M 2004 Quantum Inf. Process. 3 115, 2004 Springer-Verlag.

[31] Veronica Cerletti, W. A. Coish, Oliver Gywat and Daniel Loss, *Recipes for spin-based quantum computing*, arXiv:cond-mat/0412028 v2 24 Feb 2005.

[32] Kodera T, van der Wiel W G, Maruyama T, Hirayama Y and Tarucha S 2005 *Fabrication and Characterization of Quantum Dot Single Electron Spin Resonance Devices* (Singapore: World Scientific Publishing) at press.

[33] D. Aharonov, A. Kitaev, and N. Nisan. ‘Quantum Circuits with Mixed States’. *Proceedings of the 31st Annual ACM Symposium on the Theory of Computation (STOC’98)*, 20–30, 1998.

[34] A, Barenco, Charles H. Bennett, Richard Cleve, David P. DiVincenzo, Norman Margolus, Peter Shor, Tycho Sleator, John Smolin, and Harald Weinfurter. ‘Elementary Gates for Quantum Computation’. *Physical Review A*, 52(5):3457–3467, 1995.

[35] D, DiVincenzo. ‘Two-Bit Gates Are Universal for Quantum Computation’. *Physical Review A*, 51(2):1015–1022,1995.

[36] Richard Cleve, Artur Ekert, Leah Henderson, Chiara Macchiavello, and Michele Mosca. ‘On Quantum Algorithms’. *Complexity*, 4:33–42, 1999.

[37] J.Eisert and M.M.Wolf, *Quantum computing*, arXiv: quant-ph/0401029 v2, 28 Apr 2004.

[38] Barbara Terhal. *Quantum Algorithms and Quantum Entanglement*. Ph.D. thesis, University of Amsterdam, 1999.

[39] L.K. Grover, *A fast quantum mechanical algorithm for estimating the median,* lanl e-print quant-ph/9607024.

[40] Lov K. Grover, *A fast quantum mechanical algorithm for database search*, 3C-404A, Bell Labs.

[41] Lov K. Grover, *How fast can a quantum computer search?*, lanl e-print qut-ph/9809029.

[42] L.Grover, *Quantum computers can search arbitrary large databases by a single a query*, lanl e-print quant-ph/9706005.

[43] A, Djeloul Berkane, *Transport de spin dans des nanomatériaux application spintronique*, Magistere thesis, UHBC –Chlef-

[1] http://public.irts.net, The 2003 International Technology Roadmap for Semiconductors

[2] Shooman M., *Reliability of Computer System and Networks: Fault Tolerance,* Analysis. and Design, Wiley Interscience Publication, New York 2002.

[3] Flynn M., et al., *Deep-Submicron Microprocessor Design Isues*, IEEE Micro, Vol. 19, No. 4, 1999, pp. 11-22

[4] Plummer J., Griffin P., *Material and Process Limits in Silicon VLSI Technology*, Proceeedings of the IEEE, vol. 89, No. 3, March 2001, pp. 240-258.

[5] www.Wikipedia.com/transistors

[6] The national technology Roadmap for semiconductors, tech. report, Semiconductor Industry Assn., San Jose, Calif., 1994 and 1997 (updated).

[7] K.Nowka, High performance CMOS System Design Using Wave Pipelining, Phd thesis, Stanford Univ., Dept., Electrical Eng ., 1995.

[8] Varadarajan at al., *Low Power Design Issues*, in The Computer Engineering Handbook, V. Oklobdzija, Ed, CRC Press, Baca Raton, 2002..

[9] http://www.sematech.org/, Semiconductor Industry Association (03) The National Technology Roadmap for Semiconductors (online)

[10] Patt Y., *Requirements, Bottlenecks, and Good Fortune: Agents for Microprocessor Evolution*, Proc. of the IEEE, Vol., 89, No. 11 2001, pp.1553- 1559.

[11] Burger D., Goodman J., *Billion-Transistor Architectures: There and Back Again*, IEEE Computer, Vol. 37, No. 3, 2004, pp. 22-28.

[12] Mile Stojˇcev, Teufik Toki´c and Ivan Milentijevi, *´The limits of semiconductor technology and oncoming challenges in computer microarchitectures and architectures'* FACTA University (NIS). ser: electronic energy. vol. 17, December 2004, p 285-312

[13] www.Wikipedia.com/Sterm-Gerlach experiment

[14] N. Mott, Proc. Roy. Soc. *A .'Electron theory of transition methods'* Phys Revolution 153, 699 (1936).

[15] Stefano Sanvito *'Ab-initio methods for spin-transport at the nanoscale level'* Lund University Department of Physics, P.O. Box 1048 Blindern, 0316 Oslo; November 1998.

[16] G. Binasch, P. Gr¨unberg, F. Saurenbach, and W. Zinn, *'single- crystalline Fe/Ag/Fe nanopillars'* Phys. Rev. B 39, 4828 (1989).

[17] S. Parkin, N. More, and K. Roche, *"Magnetotransport Properties of Magnetically Soft Spin-Valve Structures"*, Phys. Rev. Lett. 64, 2304 (1990).

[18] K. Tsukagoshi, B. W. Alphenaar, and H. Ago, "*Spin coherent transport in a ferromagnetically contacted carbon nantube"*, Nature 401, 572 (1999).

[19] B. Zhao, I. M¨onch, T. M¨uhl, H. Vinzelberg, and C. Schneider, J. Appl. *'Spin transport'* Phys. 91, 7026 (2002).

[20] B. Zhao, I. M¨onch, H. Vinzelberg, T. M¨uhl, and C. Schneider, *'Quantum spin transportation in Ga/As'* Appl. Phys. Lett. 80, 3144 (2002).

[21] S. Sahoo, T. Kontos, C. Sch¨onenberger, and C. S¨urgers, *'Nanoelectronics II - Spintronics and Magnetotransport'* cond-mat/0411623 .

[22] A. Jensen, J. Nygard, and J. Borggreen, *in Toward the controllable quantum states*, Proceedings of the International Symposium on Mesoscopic Superconductivity and Spintronics, H. Takayanagi and J. Nitta , 33 (2003).

[23] Pkilip Kaye, Raymond Laflamme and Michele Mosca, *An introduction to Quantum computing*, Oxfor University Publisher, November 2006.

[24] Dirk Bouwmeester, Artur Ekert, Anton Zeilinger (Eds.), *The Physics of Quantum Information, Quantum Cryptography,Quantum Teleportation, Quantum Computation*, Springer (2000).

[25] Schleser R, Ruh E, Ihn T, Ensslin K, Driscoll D C and Gossard *'Kondo Effect in a Many-Electron Quantum Ring'*, A C 2004 Appl. Phys. Lett. 85 2005

[26] Field M, Smith C G, Pepper M, Ritchie D A, Frost J E F, Jones G A C and Hasko D G *Imaging random telegraph signal sites near a quasi 1D electron system* 1993 Phys. Rev. Lett. 70 1311

[27] Petta J R, Johnson A C, Marcus C M, Hanson M P and *Gossard Spin-dependent transport in molecular tunnel* A C 2004 Phys. Rev. Lett. 93 186802

[28] Kodera T, van der Wiel W G, Ono K, Sasaki S, Fujisawa T and Tarucha *"Lifting of spin blockade by hyperfine interaction in vertical direction"* S 2004 Physica E 22 518, 2004

[29] Lee H, Johnson J A, Speck J S and Petroff P M 2000 J. Vac. Sci. Technol. “*Transport signatures of correlated disorder in a two-dimensional electron gas”* B 18 2193

[30] Engel H-A, Kouwenhoven L P, Loss D and Marcus C M 2004 “*Quantum Information Process”*. 3 115, 2004 Springer-Verlag.

[31] Veronica Cerletti, W. A. Coish, Oliver Gywat and Daniel Loss, *Recipes for spin-based quantum computing*, arXiv:cond-mat/0412028 v2 24 Feb 2005.

[32] Kodera T, van der Wiel W G, Maruyama T, Hirayama Y and Tarucha S 2005 *Fabrication and Characterization of Quantum Dot Single Electron Spin Resonance Devices* (Singapore: World Scientific Publishing) at press.

[33] D. Aharonov, A. Kitaev, and N. Nisan. ‘Quantum Circuits with Mixed States’. *Proceedings of the 31st Annual ACM Symposium on the Theory of Computation (STOC’98)*, 20–30, 1998.

[34] A, Barenco, Charles H. Bennett, Richard Cleve, David P. DiVincenzo, Norman Margolus, Peter Shor, Tycho Sleator, John Smolin, and Harald Weinfurter. ‘Elementary Gates for Quantum Computation’. *Physical Review A*, 52(5):3457–3467, 1995.

[35] D, DiVincenzo. ‘Two-Bit Gates Are Universal for Quantum Computation’. *Physical Review A*, 51(2):1015–1022,1995.

[36] Richard Cleve, Artur Ekert, Leah Henderson, Chiara Macchiavello, and Michele Mosca. ‘On Quantum Algorithms’. *Complexity*, 4:33–42, 1999.

[37] J.Eisert and M.M.Wolf, *Quantum computing*, arXiv: quant-ph/0401029 v2, 28 Apr 2004.

[38] Barbara Terhal. *Quantum Algorithms and Quantum Entanglement*. Ph.D. thesis, University of Amsterdam, 1999.

[39] L.K. Grover, *A fast quantum mechanical algorithm for estimating the median,* lanl e-print quant-ph/9607024.

[40] Lov K. Grover, *A fast quantum mechanical algorithm for database search*, 3C-404A, Bell Labs.

[41] Lov K. Grover, *How fast can a quantum computer search?*, lanl e-print qut-ph/9809029.

[42] L.Grover, *Quantum computers can search arbitrary large databases by a single a query,* lanl e-print quant-ph/9706005.

[43] Alcalde, A. M., Q. Fanayo, and G. E. Marques, *Electron-phonon induced spin relaxation in InAs quantum dots*, 2004, Physica E 20, 228.

[44] D. Deutsch and R. Jozsa, "*Rapid Solution of Problems by Quantum Computation,"* Proc. Royal Soc. London, London, vol. 439, 1992, pp. 553–558.

[45] P.W. Shor, "*Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer,"* Proc. 35th Ann. Symp. Foundations of Computer Science, IEEE CS Press, Los Alamitos, Calif., 1994,

[46] C. Zalka, *"Using Grover's Quantum Algorithm for Searching Actual Databases,"* Physical Rev. A, vol. 62, 2000

[47] C.H. Bennett et al., *"Strengths and Weaknesses of Quantum Computing,"* SIAM J. Computing, vol. 26, no. 5, pp. 1510–1523, 1997; http://xxx.lanl.gov/archive/quant-ph/9701001(current 5 Feb. 2001)

[48] T. Hogg, B.A. Huberman, and C.P. Williams, *"Phase Transitions and the Search Problem,"* Artificial Intelligence, vol. 81, nos. 1–2, 1996, pp.1–15.

[49] N.J. Cerf, L.K. Grover, and C.P. Williams, *"Nested Quantum Search and Structured Problems,"* Physical Rev. A, vol. 61, 2000

[50] S. Aaronson. *'Quantum Computing, Postselection*, *and Probabilistic Polynomial-Time'*. Proceedings of the Royal Society of London A, 461:3473– 3482, 2005.

[51] D. Aharonov, A. Kitaev, and N. Nisan. *'Quantum Circuits with Mixed States'*. Proceedings of the 31st Annual ACM Symposium on the Theory of Computation (STOC'98), 20–30, 1998.

[52] A. Ambainis. *'Quantum Lower Bounds'* J. Comput. Syst. Sci. 64:750–767, 2002.

[53] Adriano Barenco, Charles H. Bennett, Richard Cleve, David P. DiVincenzo, Norman Margolus, Peter Shor, Tycho Sleator, John Smolin, and Harald Weinfurter. *'Elementary Gates for Quantum Computation'*. Physical Review A, 52(5):3457–3467, 1995.

[54] Michel Boyer, Gilles Brassard, Peter Høyer, and Alain Tapp, *'Tight Bounds on Quantum Searching'*. Fortschritte der Physik 56(5–5):493–505, 1988.

[55] H. Buhrman, C. Durr, M. Heiligman, P. Høyer, F. Magniez, M. Santha, and R. de Wolf. *'Quantum Algorithms for Element Distinctness'*. SIAM J. Comput., 34:1324–1330, 2005.

[56] Thomas Beth. *'Generalized Fourier Transforms'*. Trends in ComputerAlgebra, 92–118, 1987.

[57] Donny Cheung. *'Using Generalized Quantum Fourier Transforms in Quantum Phase Estimation Algorithms'*. MMath Thesis. University of Waterloo, 2002.

[58] W. van Dam, S. Hallgren, and L. Ip. *'Quantum Algorithms for Some Hidden Shift Problems'*. Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA'03), 489–498, 2003.

[59] Richard Jozsa. *'Searching in Grover's Algorithm'*. arXiv e-print quantph/9901021.

[60] A. Childs, E. Farhi, and S. Gutmann, *Quantum Inform. Process*. 1, 35 (2002).

[61] I . S. Luryi, J. M. Xu, and A. Zaslavsky, eds., *Future Trends in Microelectronics: Reflections on the Road to Nanotechnology*, NATO AS1 Series E Vol. 323, Dordrecht: Kluwer Academic, 1996.

[62] S. Luryi, J. M. Xu, and A. Zaslavsky, eds., *Future Trends in Microelectronics: The Road Ahead*, New York: Wiley Interscience, 1999.

[63] S. Luryi, J. M. Xu, and A. Zaslavsky, eds., *Future Trends in Microelectronics: The Nano Millennium*, New York: Wiley Interscience/IEEE Press, 2002.

[64] S. Luryi, J. M. Xu, and A. Zaslavsky, eds., *Future Trends in Microelectronics: The Nano, The Giga, and The Ultra*, New York: Wiley Interscience/lEEE Press, 2004.

[65] Abalmassov, V. A., and F. Marquardt, Electron-nuclei spin relaxation through phonon-assisted hyperfine interaction in a quantum dot, 2004, Phys. Rev. B 70, 75313.

[66] Abolfath, R. M., P. Hawrylak, and I. Zuti´ c, *Tailoring magnetism in quantum dots*, 2007b, Phys. Rev. Lett. 98, 207203.