UNIVERZA V LJUBLJANI
FAKULTETA ZA MATEMATIKO IN FIZIKO
ODDELEK - FIZIKA

Matej Posinković

KVANTNI RAČUNALNIKI

SEMINAR

Mentor: prof. Anton Ramšak

Ljubljana, 2003

KAZALO

I.UVOD IN POVZETEK

Kako je do ideje kvantnih računalnikov sploh prišlo? Richard Feynman je ugotovil, da je nemogoče smotrno sprogramirati simulacijo razvoja splošnega kvantnega sistema na klasičnem računalniku. Časovna zahtevnost izvedbe simulacije je bila eksponentna. Namesto da bi Feynman sprejel to kot težavo, je v tem zagledal priložnost... Rodila se je ideja kvantnega računalnika.

V pričujočem seminarju bom posvetil več pozornosti teoretični plati kvantnih računalnikov kot pa današnjim poskusom njihove fizične izvedbe. To pa zato, ker se mi zdi pomembneje povedati KAJ so kvantni računalniki kot pa KAKO so. Kajti če ne vemo KAJ želimo narediti, potem sploh ne moremo razmišljati KAKO to izvesti.

Začel bom z osnovnimi pojmi iz sveta kvantnih računalnikov in na njih zgradil zapletenejše strukture. Pri tem bom posebno pozornost namenil vprašanju kje konkretno s kvantnimi računalniki lahko pridobimo, se obregnil ob težave, ki spremljajo to temo, ter končal s poslednjimi dosežki na tem področju. Kljub temu, da bo poudarek seminarja na teoretični plati, bom poskušal s konkretnimi predstavami osmislit teorijo.

II. KUBIT

Imejmo dvoje binarnih stanj (eno stanje je sestavljeno iz treh kubitov; vsak kubit lahko ima le dve stanji: 0 ali 1):

$$011$$
$$111$$

Prvo stanje lahko (npr.) predstavlja število 3, drugo pa število 7. V splošnem lahko trije biti predstavljajo $2^3 = 8$ različnih konfiguracij. To velja tako za klasičen nabor treh bitov kot za kvantni. Vendar obstaja razlika med klasičnim in kvantnim: v klasičnem je lahko v danem trenutku s tremi (ali n) biti shranjena le ena konfiguracija (število), medtem ko lahko s tremi (n) kvantnimi biti shanimo vseh $2^3 = 8$ (2^n) kombinacij (o tem malce kasneje).

Najprej povejmo kaj je kvantni bit. Kvantni bit ali kubit je kvantni sistem, v katerem sta Booleanova 0 in 1 predstavljena s parom dveh kvantnih stanj: $|0>$ in $|1>$. Ti dve stanji sta si ortogonalni ter normalizirani. Vsako (splošno) stanje kubita je izraženo z linearno kombinacijo teh dveh stanj: $\alpha|0> + \beta|0>$ [4]. Tako lahko skupek n kubitov sestavimo v niz in ga poimenujemo kvantni register reda n. To je zametek kvantnega racunalnika. Sicer pa je (čisto konkretno) kubit mikroskopski sistem kot na primer (nuklearni) spin, polariziran foton...

In vzemimo za primer kvantni register reda 3. število 6 zapišem kot skupek naslednjih stanj $|1> \otimes |1> \otimes |0> = |110>$. Tako lahko zapišem katerokoli število od 0 pa do 7, vendar imam hkrati shranjeno le eno število. če pa se poslužim kvantne mehanike (njene lastnosti superpozicije stanj) in namesto stanja $|a>$ (a $\in$ 0,1) postavim kubit v stanje $\frac{1}{\sqrt{2}}(|0> + |1>)$ lahko s tremi kubiti shranim vse možne kombinacije (pri tem spuščam normalizacijski faktor) [4]:

$$(|1> + |0>) \otimes (|1> + |0>) \otimes (|1> + |0>) =$$
$$|000> + |001> + |010> + |011> + |100> + |101> + |110> + |111>$$

III. KVANTNA LOGIČNA VRATA IN KVANTNO OMREŽJE

Radi bi s kubiti izvajali računske operacije. Da bi to bilo mogoče, moramo najprej prevzeti popoln nadzor nad posameznim kubitom; iz poljubne linearne kombinacije stanj $|0>$ in $|1>$ moramo znati preiti v kakršnokoli novo linearno kombinacijo teh dveh osnovnih stanj. To lahko storim s pomočjo kvantnih logičnih vrat, ki na kubitu opravijo (unitarno) operacijo. V klasičnem vezju so logičnih vrat OR, AND,..., v kvantnem vezju pa imamo [4]:

- Hadamardova vrata
- faznia vrata
- C-NOT vrata

Poglejmo, kako so posamezna vrata zgrajena.

IIIa. HADAMARDOVA VRATA

Vrata so definirana z matriko:

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

ali

$$|x> \rightarrow (-1)^x |x> + |1-x>,$$

kjer je x $\in$ 0,1. Poglejmo, kaj se zgodi, če hadamardova vrata uporabimo dvakrat (pri tem uvajam tudi notacijo: "minus - crka - minus"je oznaka za kvantna logična vrata; katera vrata, je razvidno iz crke):

$$|0> -H- |1> + |0> -H- |0>$$
$$|1> -H- |1> - |0> -H- |1>$$

Hadamardova vrata se lahko uresničijo z NMR tehniko.

IIIb. FAZNA VRATA

Fazna vrata so definirana z matriko:

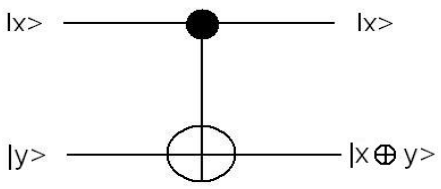
$$\mathbf{f}(\phi) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 \\ 0 & e^{i\phi} \end{pmatrix}$$

ali: $|0> \rightarrow |0>$, $|1> \rightarrow e^{i\phi}|1>$. Iz posameznega stanja (recimo $|0>$) lahko dobim s temi dvemi vrati poljubno linearno kombinacijo stanj $|0>$ in $|1>$:

$$|0> -H- f(2\theta) - H - f(\frac{\pi}{2} + \phi) - cos\theta|0> + e^{i\phi} sin\theta|1>$$

Iz posmeznega stanja lahko z hadamardovimi vrati in faznim zasukom preidem v poljubno kombinacijo dveh osnovnih stanj.

IIIc. C-NOT (ali XOR) VRATA



Slika 1: Shema C-not vrat

Zakaj torej še C-NOT vrata? Ko delamo z dvemi (ali več) kubiti poznamo dvoje vrst stanj: stanja, v katera lahko pridemo z množenjem dveh kubitov (ločena stanja), ter nerazdružljiva stanja (stanja v katera ne moremo le z množenjem dveh ali več kubitov). Primer ločenih stanj:

$$\alpha|00> +\beta|01> = |0> \otimes(\alpha|0> +\beta|1>).$$

Primer t.i. nerazdružljivih stanj je:

$$\alpha|00> +\beta|11> .$$

Takega stanja ne moremo nikakor zapisati s produktom valovnih funkcij (kot na primer v prejšnjem primeru). To pa lahko naredimo s C-NOT vrati. Matrika za taka vrata je:

$$\mathbf{C} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix},$$

ali drugače (XOR vrata: zanikajo le, če x=1):

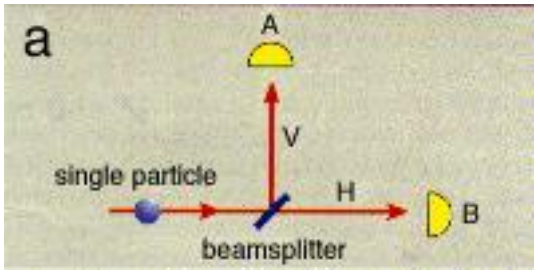
$$|x> |0> \rightarrow |x> |x> .$$

(Pri tem je moj vektor $(|00>, |01>, |10>, |11>)$.[6]) Tako lahko zdaj dobimo tudi nerazdružljiva stanja.

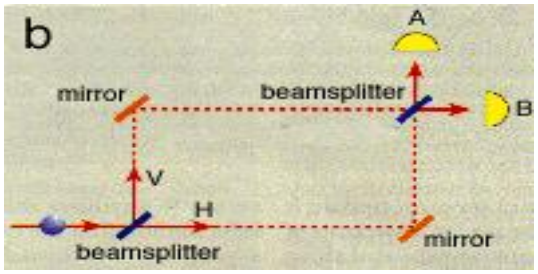
Z omenjenimi tremi vrati lahko izvedemo kakršnokoli unitarno operacijo. To troje vrat tvori univerzalni set vrat. Namesto C-NOT vrat lahko vzamemo katerakoli vrata, ki lahko naredi nerazdružljiva stanja. V splošnem so to C-U vrata, katere preslikava je: $|0> |y>$ ostane nedotaknjen,

$|1> |y>$ pa se spremeni v $|1> (U|1>)$[6].

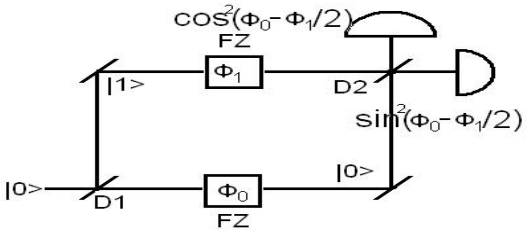
IV. PRIMER KVANTNEGA RAČUNALNIKA



Slika 2: Foton se zazna z enako verjetnostjo.



Slika 3: Detektor B ne zazna fotona.



Slika 4: Oba detektorja zaznata foton.

Pri poskusu (na Sliki 2) se vpadni (polariziran) foton razcepi na tak način, da tako detektor A kot detektor B foton zaznata z enako verjetnostjo. Razcepljeni foton v naslednjem poskusu

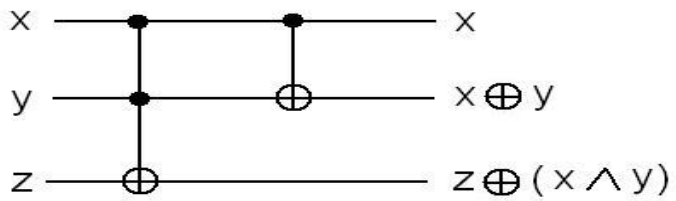
združimo (Slika 3). Rezultat: detektor B fotona ne zazna! Eksperiment še malce popravimo: na fotonovo pot dodajmo še fazni zamik (Slika 6): verjetnost zaznave fotona je odvisna od faznega zamika na obeh poteh [7]. Rezultat na sliki 3 lahko razložimo, če privzamemo naslednje: ogledalce, ki cepi foton so hadamardova vrata, fazni zamik so fazna vrata, spodnja pot predstavlja stanje $|0>$, zgornja pot pa stanje $|1>$. Poglejmo kako pridemo do rezultata:

$$|0> \rightarrow \frac{1}{\sqrt{2}}(|0> + |1>) \rightarrow \frac{1}{\sqrt{2}}(e^{i\phi_0}|0> + e^{i\phi_1}|1>) \rightarrow$$
$$\rightarrow e^{i\frac{\phi_0+\phi_1}{2}}(cos(\frac{\phi_0+\phi_1}{2})|0> + isin(\frac{\phi_0+\phi_1}{2})|1>).$$

Od tod sledi, da z različnima verjetnostima zaznamo foton na obeh detektorjih. Verjetnost zaznave na detektroju A je $sin^2\frac{\phi_0-\phi_1}{2}$ na detektorju B pa $cos^2\frac{\phi_0-\phi_1}{2}$.

V. KVANTNA ARITMETIKA

Razumeli smo qubit, vrata in mrežo vrat, radi bi naredili korak naprej in pogledali, kako računalnik računa. Vzemimo primer seštevanja. Shematično imamo vezje ponazorjeno s sliko 7. Pri tem bi izpostavil t.i. Toffoliova vrata (shema takih vrat je podana s sliko 8). Pomemben je



Slika 5: Vezje za kvantni seštevalnik

izhod na $|y>$ kubitu. Izhod je:

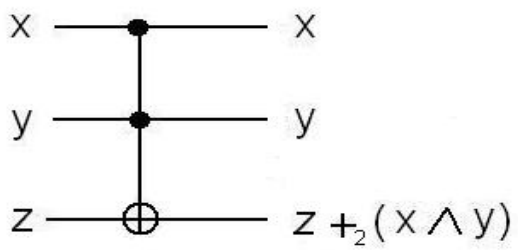
$$|xy> |z> \rightarrow |xy> |(z + (x \wedge y))mod2> .$$

Zgornjo transformacijo lahko posplošim: naj bo zato oznaka $(x_1 \wedge x_2)$ neka funkcija f(x) (x predstavlja vektor (x_1, x_2)). Torej lahko zgornjo trditev prepišem v:

$$|x, y, z> \rightarrow |x, y, (z + f(x, y))mod2> .$$

Ker lahko z različnimi postavitvami vrat sestavim različne funkcije, lahko f(x) zato razumem tudi kot neko splošno funkcijo in enačbo prepišem v neko splošno obliko [5]:

$$|x, y> \rightarrow |x, (y + f(x))mod2^m>,$$



Slika 6: Toffoliova vrata - ali dvakratni XOR: tako x kot y morata biti enaka 1, če naj se z zanika

kjer je x vektor. Ta ugotovitev nam bo kasneje pomagala pri razumevanju kvantnega algoritma.

VI. ALGORITMI

Časovna odvisnost računalniških problemov se deli na dve skupin: na P in NP skupino. Prva skupina ima polinomsko časovno zahtevnost, druga pa ne-polinomsko časovno zahtevnost. In ravno pri NP skupni problemov so kvantni računalniki tako pomembni (kajti ni nujno, da kvantni računalniki rešujejo probleme hitreje kot klasični računalniki): nekatere algoritme se da rešiti tako, da zadobijo polinomsko časovno odvisnost. Pravzaprav je bil Peter Shore tisti, ki je to prvi odkril. Problem, ki ga je obravnaval, se glasi: faktoriziraj n-bitno število. Do takrat (1994) je bila časovna zahtevnost problema $O(2^{1.9(lnn)^{1/3}(lnlnn)^{2/3}})$. Peter pa je odkril, da zmorejo kvantni računalniki problem rešiti s časovno odvisnostjo $O((lnn)^2 lnlnn(lnlnlnn))$[2]. V praksi to pomeni:

število bitov argumenta	klasični računalnik	kvantni računalnik
130	en teden	en teden
400	10^9 let	eno leto

Tu bi lahko omenil še en znan algoritem, s katerim kvantni računalniki kažejo svojo moč. To je Grooverov iskalni algoritem. Lep primer Grooverovega iskalnega algoritma je telefonski imenik (opravka imamo z neurejeno podatkovno bazo). Imamo telefonsko številko, nimamo pa priimka. S klasičnim računalnikom potrebujemo v povprečju $\frac{n}{2}$ časa, njegov kvantni mlajši brat pa to utegne že v $\log_2(n)$ [8].

Da bi pa razumeli, kje kvantni računalniki dejansko pridobijo (v primerjavi s klasičnimi računalniki) in kako pravzaprav to izvedejo si bomo v nadaljevanju ogledali potek Deutschovega algoritma.

VII. PRVI KVANTNI ALGORITEM

Prvi algoritmi na osnovi kvantnih zakonitosti, so pokazali prednost pred navadnimi računalniki. Kot primer si poglejmo t.i. Deutschov problem [1]:
Imamo funkcijo f:$0,1 \rightarrow 0,1$, ter natanko štiri različne rešitve:

$$f(0) = f(1) = 0$$
$$f(0) = f(1) = 1$$
$$f(0) = 1, f(1) = 0$$
$$f(0) = 0, f(1) = 1.$$

Ugotoviti moramo, katera od štirih rešitev je prava. Z navadnim računalnikom je očitno, da moramo izvesti dve operaciji: pogledati moramo vrednost funkcije pri 0 (f(0)), ter pogledati vrednost funkcije pri 1 (f(1)). S kvantnim računalnikom pa lahko to naredimo le z eno operacijo/meritvijo!

Imejmo stanji $|x>$ in $|u>$, kjer je $|u>$ sestavljen iz m vhodnih stanj:

$$|u>= \frac{1}{2^{m/2}} \sum_{y=0}^{2^m-1} exp(-\frac{2\pi i}{2^m} y)|y> .$$

Kot sem pa to že izpostavil v poglavju IV., pa lahko naredimo transformacijo:

$$|x>|y>\rightarrow |x>|y+f(x)> .$$

In jo tudi naredimo na stanju $|x>|u>$:

$$|x>|u>= \frac{1}{2^{m/2}}|x> \sum_{y=0}^{2^m-1} exp(-\frac{2\pi i}{2^m} y)|y>$$
$$\rightarrow \frac{1}{2^{m/2}}|x> \sum_{y=0}^{2^m-1} exp(-\frac{2\pi i}{2^m} y)|y+f(x)>$$
$$= \frac{exp(-\frac{2\pi i}{2^m} f(x))}{2^{m/2}}|x> \sum_{y=0}^{2^m-1} exp(-\frac{2\pi i}{2^m}(y+f(x)))|y+f(x)>$$
$$= exp(-\frac{2\pi i}{2^m} f(x))|x>|u> .$$

Dobili smo fazo, ki je odvisna od funkcije f! Za primer vzemimo stanje z m=1: $(|0>+|1>)$. Kot smo ravnokar videli, se zgodi naslednje:

$$|x>(|0>-|1>) \rightarrow (-1)^f(x)|x>(|0>-|1>)$$

ali

$$[(-1)^f(0)|0>+(-1)^f(1)|1>](|0>-|1>)$$

Prvi qubit je torej v primeru, da f(0)=f(1):

$$\pm(|0> + |1>),$$

v primeru, da f(0)≠f(1).

$$\pm(|0> - |1>).$$

Valovno funkcijo spustimo skozi Hadamardova vrata in izmerimo rezultat. V primeru da je f konstantna bo meritev dala $|0>$, sicer pa bomo kubit izmerili v stanju $|1>$. Tako smo konkretno spoznali prednost kvantnega računalnika.

VIII. POGOJNA KVANTNA DINAMIKA

Pogojno kvantno dinamika lahko ponazorim s preprostim primerom. Imejmo dva qubita, ki sta sklopljena z interakcijo $\sigma_z^{(1)}\sigma_z^{(2)}$ (spin-spin, dipol-dipol interakcija...), kar prinese v hamiltonjan člen:

$$V = \hbar\omega_1\sigma_z^{(1)}\sigma_z^{(2)}.$$

To pa pomeni, da se lastna (resonančna) frekvenca določenega qubita premakne ($\omega \rightarrow \omega \pm \Omega$) glede na stanje sosednjega qubita. S stališča izvedbe XOR vrat je to dobro, saj lahko nadzorovano spreminjamo vrednost izbranega qubita. Vendar pa ta interakcija prinese tudi slabe strani. Kot na primer dekoherenco.

IX. DEKOHERENCA IN REKOHERENCA

V principu nam je jasno, kako zgraditi kvantni računalnik: kvantna logična vrata sestavimo v kvantno omrežje. Toda več kot bo logičnih vrat na kupu, težje bo obvladovati medsebojne interakcije kubitov. To je tudi največji problem realizacije kvantnih računalnikov (če odmislimo čisto konkretne težavo kako tehnično realizirati stroj). Poglejmo, kako dekoherenca, čisto teoretično, vpliva na delovanje kvantnih računalnikov.

Spet izvedimo poizkus s svetlobnim računalnikom, kot je bil opisan v poglavju IV. Le da naj bosta valovni funkciji malce drugačni [7]:

$$|0, m> \rightarrow |0, m_0>$$
$$|1, m> \rightarrow |1, m_1>,$$

kjer je $|m>$ začetno stanje, $|m_0>$ in $|m_1>$ pa končni stanji okolice. Sprememba valovne funkcije okolice se spremeni zato, ker dani kubit ($|0>$ ali $|1>$) opazujemo. In si zamislimo naslednjo zaporedje dogodkov: dani kubit pošljemo skozi hadamardova vrata, nato skozi fazna vrata, ga opazujemo (npr. približamo drugi kubit), ter končno pošljemo skozi še ena hadamardova vrata. Unitarna transformacija Hadamardovih in faznih vrat nam na izbranem stanju povzroči:

$$|0>|m> \rightarrow \frac{1}{\sqrt{2}}(|0> + |1>)|m> \rightarrow \frac{1}{\sqrt{2}}(e^{i\phi/2}|0> + e^{-i\phi/2}|1>)|m> .$$

Nato tako napravljen kubit opazujemo. Zgodi se:

$$\frac{1}{\sqrt{2}}(e^{i\phi/2}|0>+e^{-i\phi/2}|1>)|m>\rightarrow \frac{1}{\sqrt{2}}(e^{i\phi/2}|0>|m_0>+e^{-i\phi/2}|1>|m_1>).$$

Stanje se, ko ga spustimo skozi še ena hadamardova vrata, spremeni v:

$$\frac{1}{\sqrt{2}}(e^{i\phi/2}|0>|m_0>+e^{-i\phi/2}|1>|m_1>)\rightarrow$$
$$\frac{1}{2}|0>(e^{i\phi/2}|m_0>+e^{-i\phi/2}|m_1>)+\frac{1}{2}|0>(e^{i\phi/2}|m_0>-e^{-i\phi/2}|m_1>).$$

Vzemimo, da sta $< m_0|m_0 >= 1$ in $< m_1|m_1 >= 1$ ter $< m_1|m_0 >$ realen. Verjetnost za stanje $|0>|m_0>$ označimo z P_0, za stanje $|1>|m_1>$ pa s P_1. Verjetnosti sta:

$$P_0 = \frac{1}{2}(1+ < m_0|m_1 > cos\phi)$$
$$P_1 = \frac{1}{2}(1- < m_0|m_1 > cos\phi).$$

Torej: dekoherenca (vplivi okolja) lahko usodno vpliva na stanje kubita, kot na primer:

- lahko nam zamenja fazo: iz $|0>+|1>\rightarrow |0>-|1>$
- lahko nam doda majhne premike: $\alpha \rightarrow \alpha+\delta$ in iz kubita $\alpha|0>$ dobimo $(\alpha+\delta)|0>$. S časom se lahko take majhne napakice seštejejo v veliko.

Neposredna posledica teh napak je precej smešna. V primeru, če bi radi vedeli ali je kakšna napaka na kubitu, ga moramo pomeriti; če pa kubit pomerimo ga zmotimo. Kubita zmotiti ne smemo, saj se pri merjenju postavi z določeno verjetnostjo v enega izmed stanj superpozicije.

Odgovor na dehoherenco je rekoherenca [3]. Rehoherence ne bom globlje razlagal; podal bom le njeno osnovno idejo. Imejmo kvantni sistem v nekem začetnem stanju $|\Psi>$ in bi ga recimo samo radi ohranili v tem stanju za nekaj časa. Pripravimo si R dodatnih kubitov, ki jih postavimo v stanje $|\Psi>$, ter vse te kubite projeciramo v simetričen podprostor, t.j. podprostor v katerem da katerakoli permutacija niza kubitov vedno enako valovno funkcijo. Izkaže se, da s tem zmanjšamo verjetnost za napako, ne da bi pri tem zmotili kubitov samih. Primer: imejmo stanje $\phi = \alpha|0>+\beta|1>$. Stanjema $|0>$ in $|1>$ dodamo dva kubita v stanju $|0>$:

$$|0>\rightarrow |000> \; |1>\rightarrow |100> .$$

Predno novo stanje shranimo ju sprojeciramo v simetrični podprostor:

$$|000>\rightarrow |\overline{000}>= (|0>+|1>)(|0>+|1>)(|0>+|1>)$$
$$|100>\rightarrow |\overline{111}>= (|0>-|1>)(|0>-|1>)(|0>-|1>).$$

In recimo, da se dekoherenca zgodi na drugem kubitu:

$$\alpha(|0>+|1>)(|0>|m_0>+|1>|m_1>)(|0>+|1>)+$$
$$\beta(|0>-|1>)(|0>|m_0>-|1>|m_1>)(|0>-|1>).$$

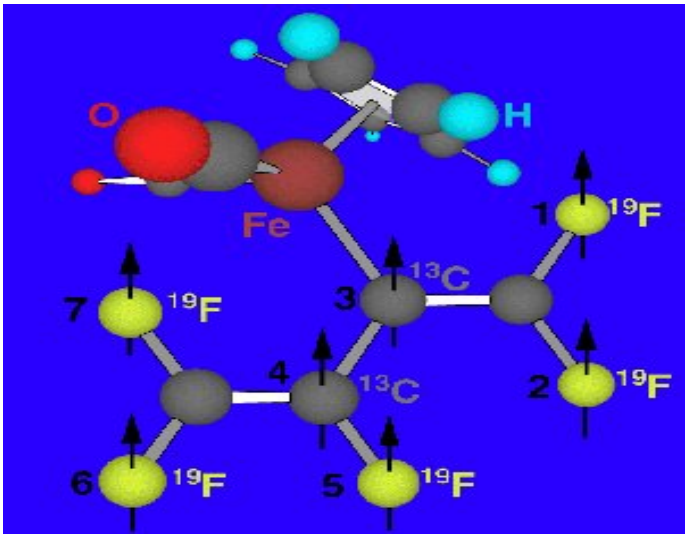
Slednje lahko prepišem kot:

$$(\alpha|\overline{000}> +\beta|\overline{111}>)(|m_0> +|m_1>) +$$
$$(\alpha|\overline{010}> +\beta|\overline{101}>)(|m_0> -|m_1>).$$

Iz slednjega lahko izluščimo, da se je zgodila dekoherenca na drugem kubitu in nato "primerno" ukrepamo.

X. KJE SMO DANES

Največji kvantni računalnik na svetu je bil predstavljen 19. decembra 2001, razvili pa so ga Kaliforniji, v enem izmed razvojnih centrov nadnacinalke IBM. Fizično je to molekula, na katero se da vplivat tako, da lahko predstavlja 7-kubitni kvantni računalnik (atomi fluora (5 kubiti) ter ogljika (2 kubita)-glej sliko 7). Z njim so izvedli Shorov faktorizacijski algoritem: število 15 jim je



Slika 7: Največji kvantni računalnik na svetu do 19.12.2001

uspelo razbiti na praštevila: 3 in 5. Kubite so vznemirjali s sunki elektromagnetnega valovanja, merjenja pa so izvajali z nuklearno magnetno resonanco (NMR). Poskus je potekal v epruvetki z 10^8 molekulami. To hkrati predstavlja tudi najzahtevnejšo operacijo izvedeno s kvantnim računalnikom do danes.

Seveda to ni prvi kvantni računalnik; zgodovina razvoja fizičnih izvedb kvantnih računalnikov gre nekako tako: leta 1998 so (na kalifornijski univerzi Berkeley) naredili 2-kubitni računalnik, nato so v IBM-ovem razvojnem centru leta 1999 s 3-kubitnim računalnikom demonstrirali Grooverov

algoritem. Zadnji dosežek sega v poletje 2000, ko se je rodil 5-kubitni stroj: tudi ta je zmogel Grooverov algoritem.

Navkljub naporom, ki jih v omenjenem raziskovalnem laboratoriju vlagajo v razvoj molekul, ki bi imele še več kubitov kot že obstoječa molekula, ni pričakovati, da bi prihodnost kvantnega računalništva ležala v tej smeri. Tako že obstajajo ideje, ki se nanašajo na elektronske spine ujete v polprevodniške nanostrukture (kvantne pike) ali elektronske in magnetne tokove v supraprevodnikih.

Vsako napovedovanje je nehvaležno; ravno tako, kot se je našel kdo, ki je zmajeval z glavo, ko so v 70 in 80 letih začeli s teorijo kvantnih računalnikov bi bile pesimistične ocene tudi tokrat neprimerne. Navkljub skromnim korakom, ki so bili do zdaj narejeni. Kot lep primer in opomin nam lahko služi razvoj informacijske tehnologije, ki je dobila v 90 letih razsežnosti, ki jih v začekih njenega razvoja niso mogli predvideti niti največji optimisti.

XI. REFERENCE

1. A. Barenco, D. Deutsch, A. Ekertand R. Jozsa, *Phys. Rev. Lett* **74** 4083 (1995)
2. P. W. Shor, *Phys. Rev.* A **54**, 1098 (1996)
3. B. Schumacher, *Phys. Rev.*A **51**, 2738 (1995)
4. Basics of Quantum Computation; Nichols A. Romeo; Massachusetts 02139; May 1998
5. Quantum Computing; Andrew Steane; University of Oxford; arXiv: quant-ph/9708022 v2 24 Sept 1997
6. Basic concepts in quantum computation; Ekert, Hayden, Inamory; University of Oxford; 16.1.2000
7. Lecture Notes for Physics 229: Quantum Information and Computation; John Preskill; California Institute of Technology; September 1998
8. Quantum Information Processing; Jaewan Kim; Dept of Physics, KAIST; 1998
9. http://xxx.lanl.gov/archive/qunt-ph
10. http://www.qubit.org