

---

---

# KVANTTIHAJASAANTIMUISTI

---

---

## LUONNONTIETEIDEN KANDIDAATIN TUTKIELMA

AXEL VAHUR

OULUN YLIOPISTO

FYSIIKAN KOULUTUSOHJELMA

TEOREETTINEN FYSIIKKA

2023

## SISÄLLYSLUETTELO

1	Tiivistelmä . . . . .	2
2	Johdanto. . . . .	2
3	Muistityyppien erot . . . . .	3
4	Klassinen hajasaantimuisti . . . . .	3
5	Kvanttimekaaniset kaksi- ja monitilasysteemit . . . . .	5
5.1	Kutritti . . . . .	6
5.2	Yhdistetyt systeemit . . . . .	6
5.3	Operaattorit . . . . .	7
6	Kvanttihajasaantimuisti . . . . .	8
6.1	Fanout-tyyppinen kvanttihajasaantimuisti . . . . .	8
6.2	Bucket brigade -tyyppinen kvanttihajasaantimuisti . . . . .	11
6.3	Kvanttilaitteiden virhealttius . . . . .	14
7	Loppupäätelmät . . . . .	15
	Viitteet . . . . .	17

# 1 TIIVISTELMÄ

Hajasaantimuisti (random access memory, RAM) on tärkeä osa klassisen tietokoneen toimintaa kaikkialla maailmassa. Hajasaantimuistia tullaan tarvitsemaan myös kvanttietokoneissa, mutta jotta ennustettuja kvanttietokoneiden mahdollistamia uskomattomia asioita voitaisi käytännössä tehdä, ei klassinen hajasaantimuisti ole riittävää. Klassisen hajasaantimuistin suora kvanttimekaaninen yleistys ei myöskään ole käytännöllinen, sillä muistiajon aikana  $O(2^n)$  laitteen reititinkomponenttia lomittuu maksimaalisesti, jossa  $n$  on muistin osoiterekisterin pituus, ja  $N = 2^n$  on muistipaikkojen määrä. Lomittuneiden komponenttien vähentämiseksi on ehdotettu niinkutsuttua bucket brigade -arkkitehtuuria, joka laskee tätä määrää eksponentiaalisesti määrään  $O(n)$ .

# 2 JOHDANTO

Tämä työ tarkastelee kvantti-hajasaantimuistia kirjallisuuskatsauksen muodossa. Kvantti-hajasaantimuistiin ja muihin kvanttilaitteisiin liittyviä ilmiöitä tutkii kvantti-informaation ala [1]. Oleellisia tähän aiheeseen liittyviä ilmiöitä ovat esimerkiksi usean kvanttimekaanisen systeemin muodostamat yhdistetyt systeemit, sekä näiden systeemien virhealttius. Tässä työssä keskitytään lähinnä kuvaamaan näiden alisysteemien, eli kubittien ja kutrittien, muodostamia yhdistettyjä systeemeitä, eikä niinkään erilaisten virheiden esiintymistodennäköisyyksiin tai virheiden korjaamiseen.

Toimivaa kvantti-hajasaantimuistia tarvitaan, jos tulevaisuudessa tullaan rakentamaan suuremman mitatakaan, eli käytännössä hyödyllisiä, kvanttietokoneita [2]. Yksittäisten laskutoimitusten tapauksessa kvanttietokoneen prosessorille voisi syöttää haluttuun tilaan ns. käsin alustettuja kubitteja ja kvantti-prosessorin laskeman ulostulon voisi tallentaa vaikka kirjoittamalla sen paperille ja myöhemmin alustaa prosessorille syötettävät kubitit tuohon samaan tilaan, kun sitä tarvitaan jossain myöhemmässä laskutoimituksessa. Siinä missä klassisen tietokoneen prosessorille voi syöttää ainoastaan klassista dataa ja sen laskemat ulostulot ovat klassisia, tulee kvanttietokoneen prosessorille voida syöttää kvanttimekaanisessa superpositiossa olevaa dataa ja sen laskeman ulostulon on oltava helposti tallennettavissa myöhempää käyttöä varten, aivan kuten klassisissakin tietokoneissa. Yksinkertaisuuden vuoksi, tässä työssä muistipaikkaan tallennetun datan oletetaan olevan klassista, jotta sitä voidaan kopioida kloonattomuusteoreemaa vioittamatta. Tästä huolimatta kvantti-hajasaantimuisti kykenee suorittamaan eri muistipaikkojen yhtäaikaista lukemista, ja niihin kirjoittamista, kvanttimekaanisessa superpositiossa normalisointiehdon säilyttäen.

Tässä työssä käydään ensin läpi pikaisesti pysyvän ja ei-pysyvän muistin erot ja siirrytään sitten klassisen hajasaantimuistin yksinkertaiseen malliin. Tämän jälkeen esitellään tarvittava kvanttimekaniikan teoria, jotta voidaan alkaa käsittelemään kvanttimekaanisia yhden ja useamman kubitin, tai kutritin, systeemeitä. Sen jälkeen siirrytään klassisen hajasaantimuistin suoraan kvanttimekaaniseen yleistyksen, eli niinkutsuttuun fanout-tyyppiseen kvanttihajasaantimustiin, ja sen tuomiin ongelmiin. Fanout-tyyppisen kvanttihajasaantimuistin paranneltua versiota, niinkutsuttua bucket brigade -tyyppistä kvanttihajasaantimustia, käydään läpi sitä seuraavassa kappaleessa. Lopuksi käydään pintapuolisesti läpi mahdollisia muistilaitteiden systeemien kokemia virheitä.

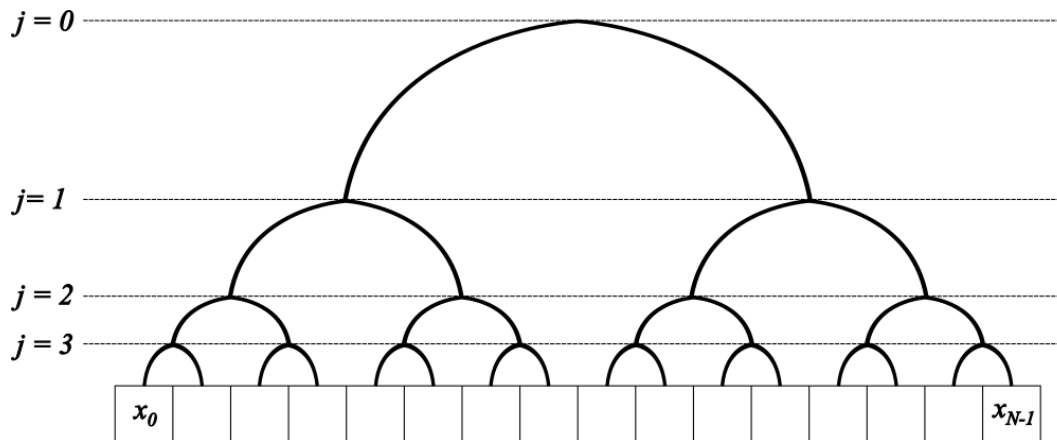
### 3 MUISTITYYPPIEN EROT

Proessori tarvitsee dataa, jolla suorittaa laskutoimituksia. Tämä data halutaan pitää tallessa ennen sen syöttämistä prosessorille, ja lisäksi prosessorin laskema tulos halutaan myös ottaa talteen, jotta se voidaan näyttää käyttäjälle, tai säästää käytettäväksi myöhemmin jossain muussa operaatioissa. Tähän tarkoitukseen on olemassa eri muistityyppejä riippuen käyttötarkoituksesta. Merkittävä ero muistityyppien välillä on pysyvällä ja ei-pysyvällä muistilla. Pysyvää muistia löytyy esimerkiksi kiintolevystä, joka säilyttää sen sisältämän datan myös sähkövirran katketessa, eli kun esimerkiksi tietokone sammutetaan. Ei-pysyvää muistia on esimerkiksi hajasaantimusti (random access memory, RAM), joka menettää sen sisältämän datan sähkövirran katketessa. Syy ei-pysyvän muistin käyttöön on kuitenkin sen nopeus. Hajasaantimuistin jokainen muistipaikka voidaan saavuttaa sen lukemista tai siihen kirjoittamista varten käytännössä yhtä nopeasti ja riippumatta siitä, mikä muistipaikka luettiin aiemmin. Perinteisessä kiintolevyssä levyä täytyy pyörittää kunnes oikea muistipaikka osuu lukupään kohdalle, ja jos sen jälkeen halutaan lukea jotain levyn toiselta puolelta, kuluu merkittävästi aikaa levyn uudelleen pyörittämiseen [3].

### 4 KLASSINEN HAJASAANTIMUISTI

Käytännön vuoksi tietokoneen hajasaantimuistin muistialkiot koostavat  $d$ -ulotteisen matriisin, jotta johtimia ei tarvita yhtä montaa kuin muistialkioita. Esimerkiksi 2-ulotteisen matriisin minkä tahansa alkion valitsemiseen tarvitaan vain oikea rivi ja sarake, eli johtimia tarvitaan yhteensä rivien ja sarakkeiden lukumäärä. Yksinkertaisimmillaan hajasaantimusti on kuitenkin vain 1-ulotteinen matriisi, eli rivi, jota nyt tarkastellaan.

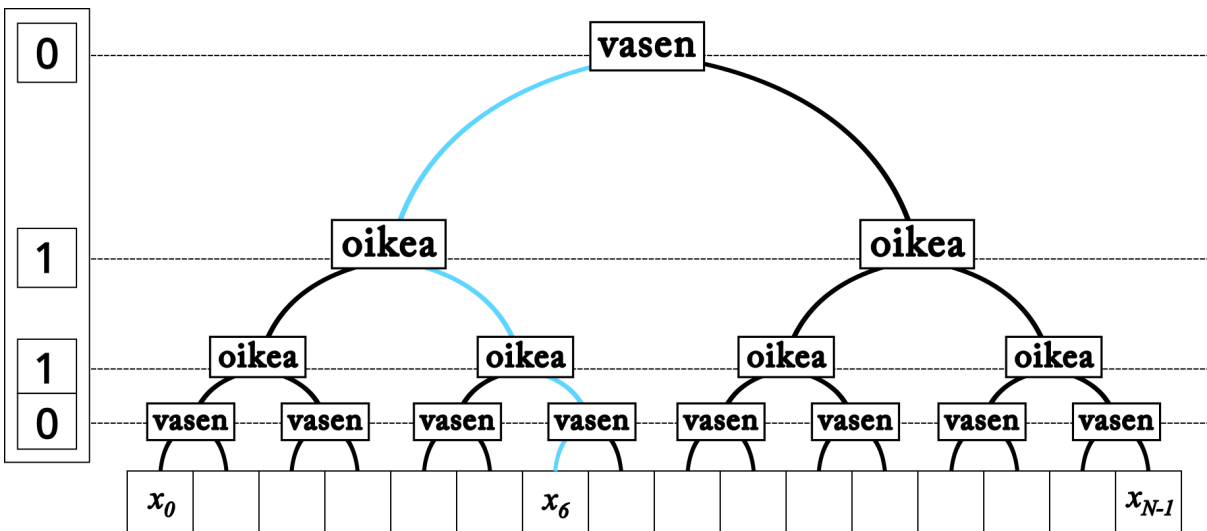
Ajatellaan, että johtimet muodostavat kaksihaararakenteen jossa on  $n$  tasoa. Ensimmäisellä tasolla on yksi risteys, joka haarautuu kahdeksi reitiksi, joista kummatkin haarautuvat edelleen kahdeksi seuraavalla tasolla [2]. Siis tasolla  $j$ , jonka laskeminen aloitetaan nolasta, on  $2^j$  risteystä. Risteysten välissä jokaisessa haarassa on transistori, joka on joko tilassa 0 tai 1. Kaksihaararakenteen pohjalla on  $2^n = N$  muistipaikkaa, joihin jokaiseen johtaa yksikäsitteinen reitti, kun peruuttamista kesken matkan ei sallita. Jokainen näistä  $N$  paikasta on muistialkio ja yhdessä ne muodostavat muistirekisterin, joka nyt on 1-ulotteinen matriisi. Kuvan 1 tapauksessa nelitasoinen kaksihaararakenteen johtaa 16-paikkaiseen muistirekisteriin.



Kuva 1. Nelitasoinen kaksihaararakenteen.

Kun halutaan päästä tiettyyn muistipaikkaan käsiksi, tulee avata reitti, joka johtaa kyseiseen muistipaikkaan. Koska risteyksestä voi kääntyä vain vasemmalle tai oikealle, riittää ajo-ohjeiksi antaa lista peräkkäisiä käännoiksi. Vasen tai oikea ovat siis ainoat kaksi tilaa, joissa risteys voi olla, eli sen kuvaamiseen voidaan käyttää bittiiä. Tarkemmin vasen tai oikea reitti avataan asettamalla vasemmassa tai oikeassa haarassa oleva transistori tilaan 1, jolloin virta kulkee vain sitä haaraa pitkin. Kuvassa 1 haararakenteen vasemmalla puolella on  $n$ -paikkainen biteistä koostuva osoiterekisteri, jonka  $j$ :s osoitebitti avaa kaikki  $j$ :nnen tason risteysten vasempien haarojen reitit mikäli osoitebitti on 0, ja vastaavasti kaikki oikeat reitit mikäli se on 1 [2].

Kuvassa 2 on esimerkiksi asetettu osoiterekisteri tilaan 0110 ja sinisellä on merkattu sen avaama yksikäsitteinen reitti muistipaikkaan  $x_6$ . Huomattavaa tässä menetelmässä eli arkkitehtuurissa on, että vaikka reitin varrella on vain neljä transistoria, aktivoidaan silti jokaista rakenteen risteystä kohti yksi transistori, siis yhteensä 15 transistoria. Selvästi risteys on aina  $N - 1 = 2^n - 1$  kappaletta, eli transistoreita aktivoidaan  $O(N)$  verran muistiajon aikana, kun muistirekisteri on 1-ulotteinen. Muistiajossa mukana olevien risteysten määrä on sama kuin aktivoitujen transistorien määrä, joten mallia voidaan yksinkertaistaa edelleen ajattelemalla risteystä reitittimenä ja unohtamalla transistorit, joita on kaksi jokaista reitintä



Kuva 2. Nelitasoiselle muistirakenteelle annettu osoite 0110 osoittaa muistipaikkaan  $x_6$ .

kohti. Yleisesti  $d$ -ulotteisen muistirekisterin muistiajossa aktivoidaan  $O(N^{1/d})$  transistoria [4], ja selvästi, kun  $d = 1$ , saadaan rivin tapauksen tulos  $O(N)$ .

Muistilaitteessa ovat lisäksi olemassa luku- ja kirjoitustilat, jotka määräävät kirjoitetaanko saavutettuun muistialkioon jokin uusi arvo vaiko kopioidaan ja palautetaan siellä ennestään oleva arvo ulostulorekisteriin, jotta sitä voidaan edelleen käyttää prosessorin laskutoimituksissa [3].

Ennen siirtymistä muistilaitteen kvanttimekaaniseen versioon on syytä käydä läpi kvanttimekaniikan aiheelle keskeisiä käsitteitä.

## 5 KVANTTIMEKAANISET KAKSI- JA MONITILASYSTEEMIT

Kvanttimekaanisen laitteen merkittävin ero klassiseen laitteeseen on sen kyky operoida sekä klassisia että kvanttimekaanisia superpositiotiloja. Laitteen tulee siis voida toimia lineaarisesti. Klassinen bitti, eli kaksitilasysteemi, voidaan yleistää kvanttimekaaniseksi kaksitilasysteemiksi, kubitiksi, joka on tilassa

$$|\Psi\rangle = a|0\rangle + b|1\rangle, \quad (1)$$

jossa  $|0\rangle$  ja  $|1\rangle$  ovat systeemin ominaistilat ja muodostavat ortonormaalien kannan [5]. Kertoimet  $a$  ja  $b$  ovat kompleksilukuja, mutta fysikaaliset tilat ovat aina normalisoituja, jolloin niille pätee

$$|a|^2 + |b|^2 = 1. \quad (2)$$

Ominaistilojen matriisiesitykset ovat

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \text{ ja } |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}. \quad (3)$$

Kubitin tilaa voidaan ajatella klassisena, jos toinen kertoimista  $a$  tai  $b$  on nolla, jolloin toinen kertoimista on välttämättä yksi, jotta normalisointiehto toteutuu. Kun kumpikin kerroin eroaa nolasta, on kubitti jossain tilojen  $|0\rangle$  ja  $|1\rangle$  lineaarikombinaatiossa eli kvanttimekaanisessa superpositiossa.

## 5.1 KUTRITTI

Kvanttimekaaninen kolmitilasysteemi on kutritti. Systeemillä on siis kolme ominaistilaa  $|0\rangle$ ,  $|1\rangle$  ja  $|2\rangle$ . Systeemi voi olla näiden ominaistilojen missä tahansa kvanttimekaanisessa superpositiossa

$$|\Psi\rangle = a|0\rangle + b|1\rangle + c|2\rangle, \quad (4)$$

joka täyttää normalisointiehdon

$$|a|^2 + |b|^2 + |c|^2 = 1. \quad (5)$$

Tässä työssä kutritin ominaistiloiksi  $|0\rangle$  ja  $|1\rangle$  on valittu kubitin vastaavat ominaistilat ja ominaistilaksi  $|2\rangle = |W\rangle$ , joka on niinkutsuttu epäaktiivinen tila, eikä sitä esiinny superpositiossa muiden ominaistilojen kanssa [2]. Siis  $|c|^2 = 1$  tai  $0$ .

## 5.2 YHDISTETYT SYSTEEMIT

Joukko systeemeitä muodostaa yhdistetyn systeemin, joka voidaan esittää näiden systeemien tilojen tensoritulona. Esimerkiksi kahden kubitin  $A$  ja  $B$ , jotka ovat normalisoiduissa tiloissa  $|\Psi_A\rangle = a|0\rangle + b|1\rangle$  ja  $|\Psi_B\rangle = c|0\rangle + d|1\rangle$ , muodostama systeemi on yhdistetyssä tilassa

$$\begin{aligned} |\Psi_A\rangle \otimes |\Psi_B\rangle &= |\Psi_A\rangle |\Psi_B\rangle = ac|0_A\rangle|0_B\rangle + ad|0_A\rangle|1_B\rangle + bc|1_A\rangle|0_B\rangle + bd|1_A\rangle|1_B\rangle \\ &= ac|00\rangle + ad|01\rangle + bc|10\rangle + bd|11\rangle, \end{aligned} \quad (6)$$

joka on edelleen normalisoitu ja sitä kutsutaan tulotilaksi. Mikäli kubitit ovat yhdistetyssä tilassa, jota ei voi esittää yksittäisten kubittien tilojen tensoritulona kuten yllä, ovat kubitit lomittuneessa tilassa. Esimerkki

kahden kubitin lomittuneesta tilasta on

$$|\Psi_{AB}\rangle = \frac{1}{\sqrt{2}}(|0_A\rangle |0_B\rangle + |1_A\rangle |1_B\rangle) = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle). \quad (7)$$

Vaikka tila on selvästi normalisoitu, tulisi kummankin kubitin  $A$  ja  $B$  olla jonkinlaisessa kvanttimekaanisessa superpositiotilassa, jotta termit  $|00\rangle$  ja  $|11\rangle$  syntyisivät, mutta silloin syntyisivät välttämättä myös "ristitermit"  $|01\rangle$  ja  $|10\rangle$  yhtälön (6) tapaan, joita ei selvästikään ole tilassa  $|\Psi_{AB}\rangle$ . Tulotilojen ja lomittuneiden tilojen ero on merkittävä, sillä lomittuneen systeemin yhden alisysteemin mittaaminen vaikuttaa muiden alisysteemien saman suureen mittaustodennäköisyyksiin [5].

Dekoherenssi tarkoittaa alisysteemin irtautumista lomittuneesta tilasta ja se voi ilmetä esimerkiksi edellä mainittuna suureen mittaamisena. Mikäli mittaaminen tapahtuu vahingossa, on kyseessä virhe, ja kvanttilaitteiden olosuhteiden on oltava sellaiset, että virheitä sattuisi mahdollisimman vähän.

### 5.3 OPERAATTORIT

Kubitin tilaa voidaan kääntää ainoastaan jollain unitarisella matriisioperaattorilla, eli matriisilla  $\hat{U}$ , joka täyttää ehdot

$$\hat{U}^\dagger = \hat{U}^{-1} \iff \hat{U}^\dagger \hat{U} = \hat{I}, \quad (8)$$

jossa  $\hat{I}$  on yksikkömatriisi. Esimerkki unitarisesta operaattorista on NOT-portti

$$\hat{X} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad (9)$$

joka on selvästi unitaarinen ja se kääntää kubitin tilan  $|0\rangle (|1\rangle) \rightarrow |1\rangle (|0\rangle)$ . Siis

$$\hat{X} |0\rangle = |1\rangle \text{ ja } \hat{X} |1\rangle = |0\rangle. \quad (10)$$

Koska kvanttimekaaniset operaattorit ovat lineaarisia, kääntyy myös kvanttimekaaninen superpositiotila

$$a |0\rangle + b |1\rangle \rightarrow \hat{X}(a |0\rangle + b |1\rangle) = \hat{X}a |0\rangle + \hat{X}b |1\rangle = a |1\rangle + b |0\rangle. \quad (11)$$

Esimerkki kahden kubitin  $A$  ja  $B$  yhdistetyn tilan unitarisesta operaattorista on kontrolloitu-NOT-portti



(controlled-NOT-portti, CNOT-portti)

$$\hat{X}_c = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \quad (12)$$

jossa ensimmäinen kubitti  $A$  on kontrolli-kubitti, joka kääntää kubitin  $B$  tilan ( $|0\rangle \rightarrow |1\rangle$  ja  $|1\rangle \rightarrow |0\rangle$ ), jos ja vain jos kontrolli-kubitin tila on  $|1\rangle$  [6]. Kahden kubitin systeemin ominaistilat ovat  $|00\rangle$ ,  $|01\rangle$ ,  $|10\rangle$  ja  $|11\rangle$ , ja ne muodostavat ortonormaalin kannan. cNOT-operaattori muuttaa esimerkiksi näitä ominaistiloja seuraavasti:

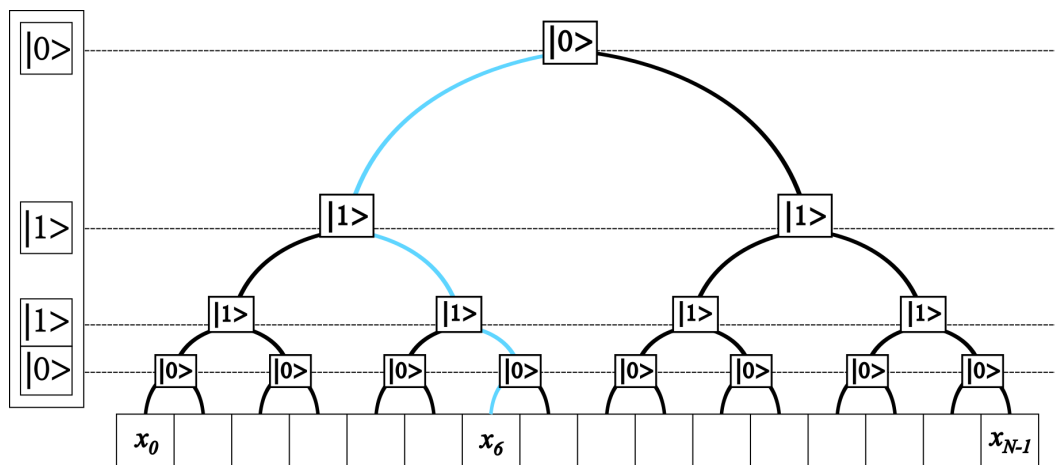
$$\begin{aligned} \hat{X}_c |00\rangle &\rightarrow |00\rangle, & \hat{X}_c |01\rangle &\rightarrow |01\rangle, \\ \hat{X}_c |10\rangle &\rightarrow |11\rangle, & \hat{X}_c |11\rangle &\rightarrow |10\rangle. \end{aligned} \quad (13)$$

## 6 KVANTTIHAJASANTIMUISTI

Klassinen hajasaantimuisti voidaan yleistää kvanttihajasaantimuistiksi, kun tietyt komponentit korvataan komponenteilla, jotka täyttävät lineaarisuuden ehdon. Tällöin muistilaitteen osoitebitit ja muistirekisterin bitit on korvattava kubiteilla ja transistoreista koostuvat reitittimet vastaavasti jollain sopivilla kvanttilaitteilla. Tällaisen muistilaitteen rakentaminen käytännössä on vaikeaa, sillä merkittävä osa laitteen komponenteista menee muistiajon aikana maksimaalisesti lomittuneeseen tilaan, joka on hyvin altis virheille [2, 4]. Käydään seuraavaksi läpi klassisen hajasaantimuistin arkkitehtuurin suoraa kvanttimekaanista yleistystä, eli niinkutsuttua fanout-tyyppistä kvanttihajasaantimuistia, sekä sille ehdotettua paranneltua versiota, eli niinkutsuttua bucket brigade -tyyppistä kvanttihajasaantimuistia.

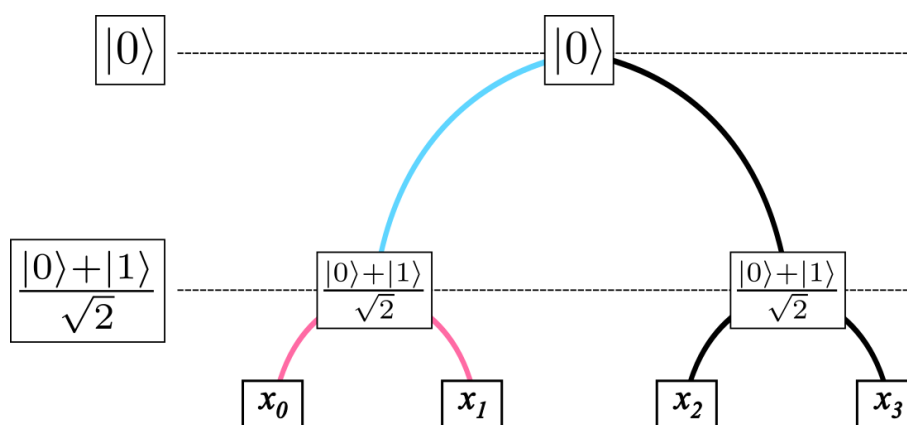
### 6.1 FANOUT-TYYPPINEN KVANTTIHAJASANTIMUISTI

Fanout-arkkitehtuurissa reitittimet sekä osoite- ja muistirekisterin bitit on korvattu kubiteilla ja toimintaperiaate on täysin sama kuin klassisen hajasaantimuistin tapauksessa:  $j$ :s osoitekubitti  $|0\rangle$  aktivoi  $j$ :nnen tason kaikkien reitittimien vasemmat haarat ja vastaavasti  $|1\rangle$  oikeat haarat. Kuvassa 3 on esimerkki



Kuva 3. Nelitasoiselle fanout-tyyppiselle kvanttihajasaantimuistille annettu osoite  $|0110\rangle$  osoittaa muistipaikkaan  $x_6$ .

klassisesta osoitteesta  $|0110\rangle$ , joka avaa yksikäsitteisen reitin muistipaikkaan  $x_6$  täysin samalla tavalla kuin klassinen muistilaite. Reittiä pitkin lähetetään sitten tilaan  $|0\rangle$  alustettu kvanttibussi, jonka tila kääntyy muistipaikan  $x_6$  sisältämän muistialkion  $|x_6\rangle$  mukaan. Yksinkertaisuuden vuoksi, oletetaan muistialkioiden olevan yhden kubitin pituisia, jolloin kvanttibussin tila voidaan kääntää cNOT-portin  $\hat{X}_c$  avulla, jossa muistialkio toimii kontrolli-kubittina. Tässä esimerkissä kaikkien kubittien, kvanttibussi mukaan lukien, yhdistetyssä tilassa ei myöskään esiinny lainkaan lomittuneita alisysteemeitä. Lomittumista alkaa esiintyä vasta, kun klassisen osoitteen sijaan annetaan jokin osoitteiden kvanttimekaaninen superpositio. Tarkastellaan tätä kahden osoitekubitin kokoisessa haararakenteessa, kuten kuvassa 4.



Kuva 4. Kaksitasoinen fanout-muistirakenne, jolle annettu osoite on kvanttimekaanisessa superpositiotilassa  $|0\rangle \left( \frac{|0\rangle+|1\rangle}{\sqrt{2}} \right)$ .

Ensimmäisen osoitekubitin tila on  $|0\rangle$  ja se avaa vasemman haaran reitin tuttuun tapaan. Toisen osoiteku-

bitin tila on  $\frac{|0\rangle+|1\rangle}{\sqrt{2}}$ , joka nyt avaa yhtäsuuren reitin vasempaan ja oikeaan haaraan, osoittaen lopulta yhtä paljon muistipaikkoihin  $x_0$  ja  $x_1$ . Muodostetaan tämän osoiterekisteristä, tilaan  $|0\rangle$  alustetuista reitittimistä sekä kvanttibussista koostuvan yhdistetyn systeemin tilan yhtälö

$$\begin{aligned}
|\Psi_{\text{fan}}\rangle &= |\Psi_{\text{osoite}}\rangle |\Psi_{\text{reitittimet}}\rangle |\Psi_{\text{bussi}}\rangle \\
&= |0_{j0}\rangle \left( \frac{|0_{j1}\rangle + |1_{j1}\rangle}{\sqrt{2}} \right) |0_{r1}0_{r2}0_{r3}\rangle |0_{\text{b}}\rangle \\
&= \frac{1}{\sqrt{2}} \left( |0_{j0}0_{j1}\rangle + |0_{j0}1_{j1}\rangle \right) |0_{r1}0_{r2}0_{r3}\rangle |0_{\text{b}}\rangle \\
&= \frac{1}{\sqrt{2}} \left( |0_{j0}0_{j1}\rangle |0_{r1}0_{r2}0_{r3}\rangle |0_{\text{b}}\rangle + |0_{j0}1_{j1}\rangle |0_{r1}0_{r2}0_{r3}\rangle |0_{\text{b}}\rangle \right), \tag{14}
\end{aligned}$$

jossa osoitekubitien  $j$  laskeminen aloitetaan nolasta ylhäältä alas ja reitittimien  $r$  laskeminen yhdestä ylhäältä alas ja vasemmalta oikealle. Tässä vaiheessa osoitekubitit eivät ole vielä kääntäneet reitittimien tiloja ja koko systeemin tilan yhtälö (14) on sievennetty muotoon, joka kuvaa selkeämmin seuraavaksi avattavia reittejä osoitteiden kvanttimekaanisen superposition termien  $\frac{1}{\sqrt{2}} |0_{j0}0_{j1}\rangle$  ja  $\frac{1}{\sqrt{2}} |0_{j0}1_{j1}\rangle$  mukaan. Kummassakin termissä ensimmäinen osoitekubitti on  $|0_{j0}\rangle$ , koska tämän osoitekubitin tila on klassinen, eikä kvanttimekaaninen superpositiotila. Seuraavaksi osoitekubitit kääntävät jokaisen niiden tason reitittimen samaan tilaan kuin itse osoitekubitti, kuten kuvassa 4:

$$|\Psi'_{\text{fan}}\rangle = \frac{1}{\sqrt{2}} \left( |0_{j0}0_{j1}\rangle |0_{r1}0_{r2}0_{r3}\rangle |0_{\text{b}}\rangle + |0_{j0}1_{j1}\rangle |0_{r1}1_{r2}1_{r3}\rangle |0_{\text{b}}\rangle \right). \tag{15}$$

Yhtälössä (15) reitittimien tilan  $|0_{r1}0_{r2}0_{r3}\rangle$  kaksi ensimmäistä reititintä  $r1$  ja  $r2$  kuvaavat siis jatkuvaa, eli katkeamatonta, reittiä muistipaikkaan  $x_0$ . Termissä on mukana myös kolmas reititin  $r3$ , joka avaa osittaisen eli ei-jatkuvan reitin reitittimestä  $r3$  muistipaikkaan  $x_2$ . Jälkimmäisen termin reitittimien tila  $|0_{r1}1_{r2}1_{r3}\rangle$  avaa vastaavasti jatkuvan reitin muistipaikkaan  $x_1$  ja osittaisen reitin muistipaikkaan  $x_3$ . Huomattavaa tässä vaiheessa on, että kaikki toisen tason ( $j = 1$ ) reitittimet ovat lomittuneessa tilassa keskenään ja tason osoitekubitin kanssa:

$$|\Psi'_{\text{fan}}\rangle = \frac{1}{\sqrt{2}} |0_{j0}\rangle |0_{r1}\rangle \left( |0_{j1}\rangle |0_{r2}0_{r3}\rangle + |1_{j1}\rangle |1_{r2}1_{r3}\rangle \right) |0_{\text{b}}\rangle, \tag{16}$$

koska ensimmäinen osoitekubitti tilassa  $|0_{j0}\rangle$ , ensimmäinen reititin tilassa  $|0_{r1}\rangle$  ja kvanttibussi tilassa  $|0_{\text{b}}\rangle$  voidaan ottaa koko systeemin  $|\Psi'_{\text{fan}}\rangle$  yhteiseksi tekijäksi, eli toisinsanoen esittää yksittäisten alisysteemien tensoritulona, jolloin ne eivät ole keskenään lomittuneita. Tason  $j = 1$  osoitekubitin ja reitittimien yhdistetty tila  $\frac{1}{\sqrt{2}} \left( |0_{j1}0_{r2}0_{r3}\rangle + |1_{j1}1_{r2}1_{r3}\rangle \right)$  taas on selvästi lomittunut. Tässä vaiheessa tilassa  $|0_{\text{b}}\rangle$  oleva kvanttibussi lähetetään haararakenteen läpi pitkin avattujen reittien superpositiota. Matkan varrella reitittimet eivät käännä kvanttibussin tilaa, joten se pysyy tilassa  $|0\rangle$ . Kun kvanttibussi saavuttaa muistipaikan  $x_0$  pitkin “ $\frac{1}{\sqrt{2}}$  vasen-vasen”-reittiä, kääntää muistialkio kvanttibussin tilaan  $|x_0\rangle$  ja vastaavasti

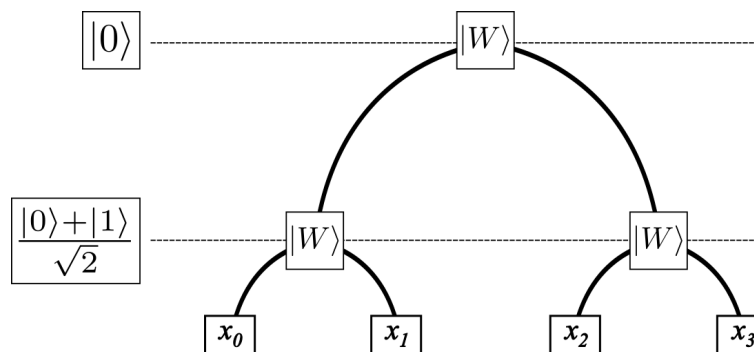
pitkin “ $\frac{1}{\sqrt{2}}$ vasen-oikea”-reittiä kvanttibussi kääntyy tilaan  $|x_1\rangle$ :

$$|\Psi''_{\text{fan}}\rangle = \frac{1}{\sqrt{2}} |0_{j0}\rangle |0_{r1}\rangle \left( |0_{j1}\rangle |0_{r2}0_{r3}\rangle |x_0\rangle + |1_{j1}\rangle |1_{r2}1_{r3}\rangle |x_1\rangle \right). \quad (17)$$

Kvanttibussi on nyt lomittunut tasolla  $j = 1$  sen osoitekubitin ja jokaisen reitittimen kanssa, koska tämä osoitekubitti on kvanttimekaanisessa superpositiotilassa. Yleisemmin, haararakenteen jokainen superpositiotilassa oleva osoitekubitti ja sitä myötä jokainen tällaisen osoitekubitin kontrolloima reititin on keskenään lomittuneessa tilassa, kuin myös lopulta rakenteen läpi lähetetty kvanttibussi [2]. Lomittuneiden, tai tarkemmin maksimaalisesti lomittuneiden, reitittimien lukumäärä kasvaa kuin  $O(N)$ , jossa  $N = 2^n$  ja  $n$  on osoitekubittien lukumäärä, aivan kuten klassisen haararakenteen tapauksessa. Reitittämiä on yleisesti  $N - 1$  kappaletta, joten enimmillään maksimaalisesti lomittuneita reitittämiä on  $N - 1$  kappaletta, mikäli jokainen osoitekubitti on superpositiotilassa.

Tässä kohtaa on hyvä alkaa miettimään, mitä seuraa jos reitittimen tilassa tapahtuu jokin virhe. Selvästikin muutos avattujen reittien varrella vaikuttaa mihin muistialkioihin reitit johtavat ja millä kertoimilla, ja muutos osoitekubitissa muuttaisi sen osittain avaamaa reittiä. Mutta myös tämän esimerkin kolmas reititin, joka ei ole avattujen reittien varrella, on maksimaalisesti lomittuneessa tilassa reitittimen  $r_2$  ja toisen osoitekubitin kanssa, jolloin muutos sen tilassa vaikuttaa kaikkiin lomittuneisiin alisysteemeihin ja siten luultavasti pilaa koko muistiajon [4]. Lomittuneiden alisysteemien vähentämiseksi on ehdotettu niinkutsuttua bucket brigade -tyyppistä kvanttihaajasaantimuistia, jota käydään läpi seuraavaksi.

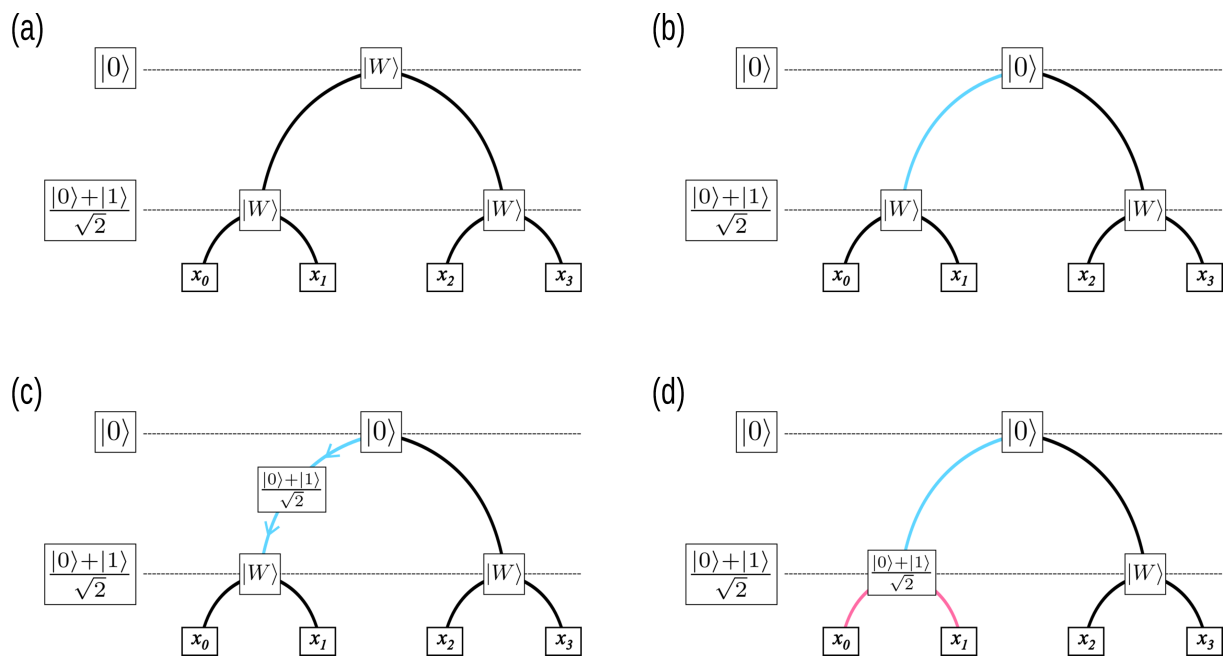
## 6.2 BUCKET BRIGADE -TYYPPIINEN KVANTTIHAJASAANTIMUISTI



Kuva 5. Kahden osoitekubitin kokoisen bucket brigade -tyyppisen kvanttihaajasaantimuistin reititinkutritit on alustettu tilaan  $|W\rangle$  ennen muistiajoa.

Bucket brigade -arkkitehtuurissa reititinkubitit on korvattu kvanttimekaanisilla kolmitilasysteemeillä,

kuten kuvassa 5, eli kutriteilla [2]. Näiden kutrittien ominaistilat ovat tutut  $|0\rangle$  ja  $|1\rangle$ , ja lisäksi myös epäaktiivinen ominaistila  $|W\rangle$ , jotka muodostavat ortonormaalien kannan. Tämä muutos yksinään ei vähentäisi lomittuneiden alisysteemien määrää, vaan sen lisäksi osoitekubitit eivät muuta kaikkien tason reititinkutrittien tilaa. Sen sijaan ensimmäinen osoitekubitti kääntää ensimmäisen reitittimen tilan kuin fanout-arkkitehtuurissa, jonka jälkeen toinen osoitekubitti lähetetään pitkin ensimmäistä haaraa kunnes se kohtaa tilassa  $|W\rangle$  olevan reitittimen, jolloin osoitekubitti kääntää tämän reitittimen tilan osoitekubitin tilaksi. Loput osoitekubitit lähetetään samaan tapaan yksitellen pitkin siihen mennessä avattua reittiä tai reittien kvanttimekaanista superpositiota, jolloin ne lopulta avaavat yksikäsitteisen reitin tai reittien kvanttimekaanisen superposition muistipaikkoihin. Tätä toimintaa on havainnollistettu kuvassa 6.



Kuva 6. Kahden osoitekubitin kokoisen bucket brigade -muistirakenteen toiminta muistiajon aikana osoitteella  $|0\rangle \left( \frac{|0\rangle + |1\rangle}{\sqrt{2}} \right)$ .

Tämän esimerkin osoite on sama kuin fanout-esimerkin tapauksessa, jotta huomataan erot lomittuneiden alisysteemien lukumäärässä. Muodostetaan tämän systeemin tilan  $|\Psi_{bb}\rangle$  yhtälö samaan tapaan kuin

aiemmin:

$$\begin{aligned}
|\Psi_{bb}\rangle &= |\Psi_{osoite}\rangle |\Psi_{reitittimet}\rangle |\Psi_{bussi}\rangle \\
&= |0_{j0}\rangle \left( \frac{|0_{j1}\rangle + |1_{j1}\rangle}{\sqrt{2}} \right) |W_{r1}W_{r2}W_{r3}\rangle |0_b\rangle \\
&= \frac{1}{\sqrt{2}} \left( |0_{j0}0_{j1}\rangle + |0_{j0}1_{j1}\rangle \right) |W_{r1}W_{r2}W_{r3}\rangle |0_b\rangle \\
&= \frac{1}{\sqrt{2}} \left( |0_{j0}0_{j1}\rangle |W_{r1}W_{r2}W_{r3}\rangle |0_b\rangle + |0_{j0}1_{j1}\rangle |W_{r1}W_{r2}W_{r3}\rangle |0_b\rangle \right) \tag{18}
\end{aligned}$$

$$|\Psi'_{bb}\rangle = \frac{1}{\sqrt{2}} \left( |0_{j0}0_{j1}\rangle |0_{r1}0_{r2}W_{r3}\rangle |0_b\rangle + |0_{j0}1_{j1}\rangle |0_{r1}1_{r2}W_{r3}\rangle |0_b\rangle \right) \tag{19}$$

$$|\Psi''_{bb}\rangle = \frac{1}{\sqrt{2}} \left( |0_{j0}0_{j1}\rangle |0_{r1}0_{r2}W_{r3}\rangle |x_0\rangle + |0_{j0}1_{j1}\rangle |0_{r1}1_{r2}W_{r3}\rangle |x_1\rangle \right). \tag{20}$$

Yhtälössä (20) kvanttibussin tila kääntyy muistialkioiden  $|x_0\rangle$  ja  $|x_1\rangle$  mukaan. Tarkastellaan sitten tämän systeemin tilan lomittuneita alisysteemeitä. Tila  $|\Psi''_{bb}\rangle$  voidaan esittää muodossa

$$|\Psi''_{bb}\rangle = \frac{1}{\sqrt{2}} |0_{j0}\rangle |0_{r1}\rangle \left( |0_{j1}\rangle |0_{r2}\rangle |x_0\rangle + |1_{j1}\rangle |1_{r2}\rangle |x_1\rangle \right) |W_{r3}\rangle, \tag{21}$$

josta huomataan, että kolmas reititin  $r3$ , edelleen tilassa  $|W\rangle$ , ei ole lomittunut muiden alisysteemien kanssa. Myöskään ensimmäinen osoitekubitti  $j0$  eikä sen ohjaama ensimmäinen reititin  $r1$  ole lomittuneita. Esimerkit ovat olleet tähän asti vain kaksitasoisia rakenteita selkeyden vuoksi, mutta sääntönä on, että ainoastaan kvanttimekaanisten superpositio-reittien varrella olevat reitittimet ja niitä ohjaavat osoitekubitit ovat lomittuneita. Kuten tästä esimerkistä huomataan, klassisessa tilassa oleva reititin  $r1$  ei ole lomittunut vaikka reitit sen jälkeen haarautuvat kahdeksi superpositio-reitiksi.

Aktiivisia reitittimiä on nyt kaikenkaikkiaan kaksi kappaletta ja kummankin reitin varrella on kaksi aktiivista reititintä. Sääntönä on kuitenkin myös, että ainoastaan reitin varrella olevat reitittimet ovat muistiajossa aktiivisia. Jos siis annetaan osoitteeksi  $\left( \frac{|0_{j0}\rangle + |1_{j0}\rangle}{\sqrt{2}} \right) \left( \frac{|0_{j1}\rangle + |1_{j1}\rangle}{\sqrt{2}} \right)$ , kääntyvät reitittimet tilaan

$$\frac{1}{2} \left( |0_{r1}0_{r2}\rangle + |0_{r1}1_{r2}\rangle + |1_{r1}0_{r3}\rangle + |1_{r1}1_{r3}\rangle \right), \tag{22}$$

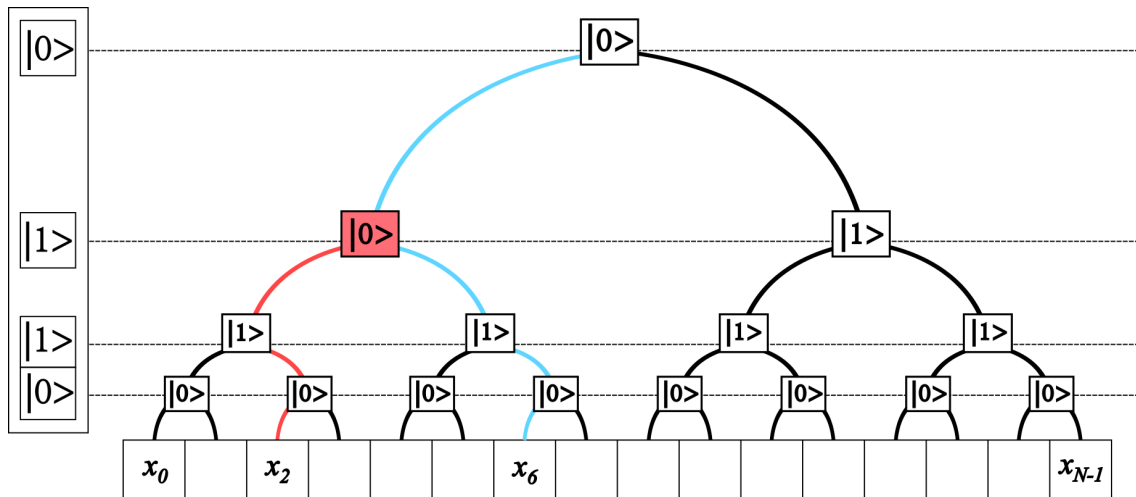
eli kaikki neljä mahdollista reittiä avautuvat kertoimella  $\frac{1}{2}$  ja kaikki kolme reititintä ovat aktiivisia, mutta jokaisen yksittäisen reitin varrella on edelleen vain kaksi aktiivista reititintä. Jos oletetaan ainoastaan aktiiviset reitittimet virhealttiiksi, sanottaisiin edelleen, että yhden muistialkion lukemisessa on ainoastaan kaksi virhealtista reititintä, vaikka yhteensä aktiivisia reitittimiä on kolme kappaletta [4]. Näin voidaan sanoa, kun otetaan huomioon reitittimen vaikutus koko muistiajoon. Tässä tapauksessa ensimmäinen reititin  $r1$  on osana jokaista neljää reittiä, joten se lasketaan yhdeksi kokonaiseksi aktiiviseksi reitittimeksi. Reititin  $r2$  on osana kahta reittiä muistipaikkoihin  $x_0$  ja  $x_1$ , eli yhteensä puoliin kaikista reiteistä, jolloin se lasketaan puolikkaaksi aktiiviseksi reitittimeksi. Vastaavasti  $r3$  johtaa loppuihin kahteen muistipaikkaan, joten tasolla  $j = 1$  on yhteensä yksi aktiivinen reititin ja niitä on kaikenkaikkiaan kaksi kappaletta.

### 6.3 KVANTTILAITTEIDEN VIRHEALTTIUS

Alunperin artikkelissa [2] oletettiin tilassa  $|W\rangle$  olevien kuitrittien olevan virhevapaita. Tämä ei kuitenkaan ole realistinen oletus, tai se vähintäänkin rajoittaa merkittävästi käytettäviä komponentteja [4, 7]. Virheiden korjaaminen on välttämätöntä kvantti-informaation sovelluksissa, jonka vuoksi erilaisia virheitä ja niiden vaikutuksia kvanttilaitteisiin on ensin mallinnettava. Yksi mahdollisista virheistä on bittikäntö-virhe, jossa yhden kubitin tila kääntyy NOT-operaattorilla  $\hat{X}$  kaavan (9) mukaan [6]

$$|0\rangle \rightarrow \hat{X} |0\rangle = |1\rangle. \quad (23)$$

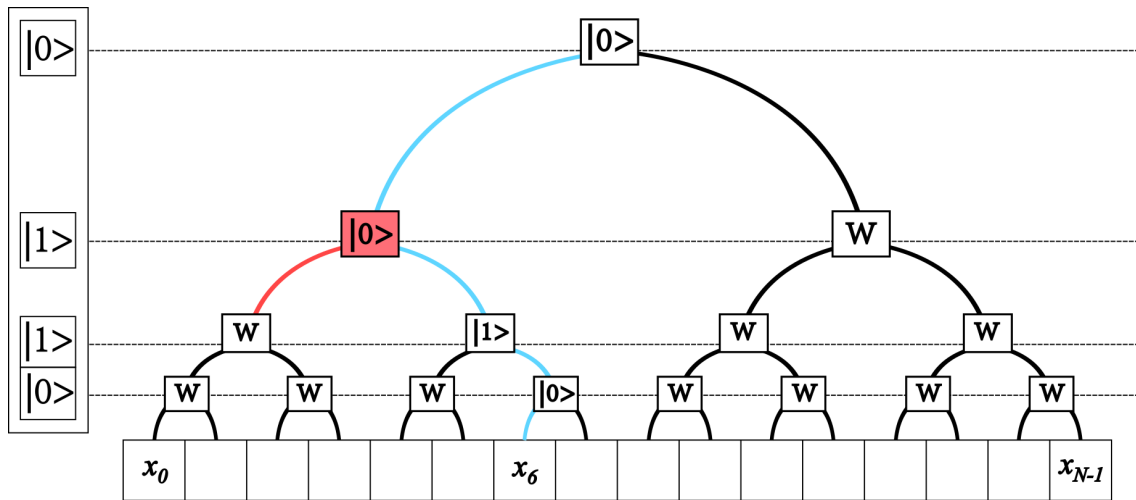
Tällainen virhe vaikuttaa eri tavoin fanout- ja bucket brigade -tyyppisissä muistirakenteissa. Fanout-arkkitehtuurissa minkä tahansa avatun reitin varrella olevan reititinkubitin kääntyminen avaa reitin väärään muistipaikkaan tai niiden superpositioon, koska kaikki reitittimet ovat aktiivisia ja ovat avanneet osittaisia reittejä pitkin muistirakennetta. Esimerkiksi kuvan 7 tapauksessa, jos tason  $j = 1$  vasemmanpuoleinen punaisella merkattu reititin  $r2$  kääntyy tilasta  $|1\rangle$  tilaan  $|0\rangle$ , jatkuisi reitti alemmilla tasoilla ensin oikeaa ja sitten vasenta haaraa, eli punaista reittiä, pitkin kuten ilman virhettä, johtaen lopulta muistipaikkaan  $x_2$ .



Kuva 7. Fanout-tyyppisessä kvanttihajasaantimuistissa tapahtuu bittikäntö-virhe toisen tason reitittimessä  $r2$ .

Bucket brigade -arkkitehtuurissa merkitystä on myös paikalla ja ajanhetkellä, jolloin tämä sama virhe tapahtuu. Kuvassa 8 on kuvan 7 esimerkitapaus esitettyä bucket brigade -muistirakenteessa, jossa kaikki paitsi onnistuneesti avatun reitin varrella olevat reititinkutritit ovat tilassa  $|W\rangle$ . Jos reitin varrella olevan jonkin ylemmän tason reitittimen tila kääntyy, kuten esimerkiksi reitittimen  $r2$ , katkeaisi reitti seuraavaan  $|W\rangle$ -tilassa olevaan reitittimeen pitkin punaista haaraa eikä kvanttibussi pääsisi kulkemaan mihinkään

muistipaikkaan [7]. Jos alin reititin tasolla  $j = 3$  kääntyy tilasta  $|0\rangle$  tilaan  $|1\rangle$ , pääsee kvanttibussi kulkemaan väärään muistipaikkaan  $x_7$  ja suorittamaan muistiajon muuten onnistuneesti.



Kuva 8. Bucket brigade -tyyppisessä kvantti-hajasaantimuistissa tapahtuu bittikääntö-virhe toisen tason reitittimessä  $r_2$  osoitekubitien onnistuneen syöttämisen jälkeen. Kuvassa  $W = |W\rangle$ .

Toinen merkittävä virhe on niinkutsuttu amplitude damping -virhe, jossa systeemi spontaanisti emittoi fotonin ja romahtaa virittyneestä tilasta perustilaan  $|1\rangle$ . Kubitin perustila on  $|0\rangle$ , joten esimerkiksi tila  $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$  romahtaisi virheen tapahtuessa tilaan  $|0\rangle$ , tuhoten superpositiotilan ja vaikuttaen kubitin kanssa lomittuneisiin alisysteemeihin [6]. Tässä työssä kutritin perustila on  $|W\rangle$ , kuten julkaisun [2] alustavassa mallissa, ja tilat  $|0\rangle$  ja  $|1\rangle$  ovat virittyneitä tiloja. Kutritin ollessa jossain virittyneessä eli aktiivisessa tilassa ja amplitude damping -virheen tapahtuessa emittoi se fotonin ja romahtaa tilaan  $|W\rangle$ . Tila  $|1\rangle$  voi myös romahtaa ensin tilaan  $|0\rangle$ , mutta joka tapauksessa superposition romahtaminen vaikuttaa systeemin kanssa lomittuneisiin systeemeihin ja vaikuttaa siten muistiajon onnistumiseen.

## 7 LOPPUPÄÄTELMÄT

Tässä työssä käytiin läpi klassisen ja kvanttimekaanisen hajasaantimuistin yhtäläisyyksiä ja eroja. Klassisen hajasaantimuistin suora kvanttimekaaninen yleistys johtaa  $O(N)$ :ään maksimaalisesti lomittuneita reititimiä yhdessä osoitekubitien kanssa, jossa  $N$  on muistipaikkojen lukumäärä, joka on erittäin virhealtis systeemi. Bucket brigade -tyyppinen kvantti-hajasaantimuisti laskee lomittuneiden reitittimien määrän  $O(\log_2 N)$ :ään, koska aktiivisia reititinkutritteja on ainoastaan avattujen superpositioreittien varrella, ja



mitä syvemmällä haararakenteessa reititin sijaitsee, sen vähemmän lomittunut se on muiden reitittimien ja osoitekubittien kanssa. Näiden kahden muistirakenteen komponenttien, eli alisysteemien, muodostamia yhdistettyjä systeemeitä havainnollistettiin kuvien ja esimerkkien avulla. Lopuksi käytiin läpi esimerkkeinä bittikäntö- ja amplitude damping -virheet, jotka kubitti tai kutritti voi kokea muiden virheiden lisäksi. Kvanttihajasaantimuistin tulevaisuus näyttää hyvinkin lupaavalle bucket brigade -arkkitehtuurin vuoksi, sillä se on vastustuskykyinen myös realistisille virhemalleille, kuten artikkelin [4] lopussa todetaan. Tästä syystä näkisin, aivan kuten artikkelin kirjoittajakin, että lähitulevaisuudessa tämän tyyppisiä kokeellisia muistilaitteita voitaisiin rakentaa laboratorioissa.

- 
- [1] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, The Edinburgh Building, Cambridge CB2 8RU, UK, 2010) 10th Anniversary ed.
  - [2] V. Giovannetti, S. Lloyd, and L. Maccone, Quantum random access memory, *Physical review letters* **100**, 160501 (2008).
  - [3] R. C. Jaeger and T. N. Blalock, *Microelectronic Circuit Design*, 5th ed. (McGraw-Hill Education, 2 Penn Plaza, New York, 2016).
  - [4] C. T. Hann, G. Lee, S. Girvin, and L. Jiang, Resilience of quantum random access memory to generic noise, *PRX Quantum* **2**, 020311 (2021).
  - [5] M. Silveri, M. Alatalo, and J. Tuorila, Lecture notes: Quantum mechanics 2, University of Oulu (2022).
  - [6] M. Silveri, Lecture notes: Quantum information, University of Oulu (2022).
  - [7] S. Arunachalam, V. Gheorghiu, T. Jochym-O'Connor, M. Mosca, and P. V. Srinivasan, On the robustness of bucket brigade quantum ram, *New Journal of Physics* **17**, 123010 (2015).