

Kvantni celični avtomati in kvantno procesiranje

Uredila dr. Miha Janež in prof. dr. Miha Mraz

oktober 2024

Predgovor

S povečevanjem gostote integriranih vezij postajajo tranzistorji vse manjše nanoelektronske naprave. Želeno delovanje tranzistorjev velikosti le nekaj nanometrov pa pričnejo ovirati kvantni pojavi in sile med elektroni. Zato so se pojavile zamisli o novih platformah, ki jih ti pojavi ne le ne bi ovirali, pač pa bi jih platforme izkoristile za procesiranje. Poleg tega lahko nove nanoelektronske naprave prinesejo še več prednosti pred današnjimi računalniki, npr. večjo gostoto, manjšo porabo energije, hitrejše delovanje itd.

Ena od takih predlaganih platform so kvantni celični avtomati (Quantum Cellular Automata - QCA). Namesto tranzistorjev vsebujejo celice z elektroni, katerih razporeditev znotraj celice določa logično stanje. Procesiranje se izvede z ustrezno razporeditvijo QCA celic, pri čemer se izkoriščata Coulombova sila in kvantno tuneliranje. Do sedaj so bile fizično implementirane le manjše QCA strukture z majhnim številom celic, ker je praktično težko izdelati QCA celice nanometrskih velikosti. Možno pa je načrtovati in optimizirati večja QCA vezja s pomočjo računalniškega simulatorja.

V zadnjem času je zaradi praktične implementacije veliko pozornosti posvečene kvantnemu procesiranju z osnovnimi gradniki kvantnimi biti. Superpozicija in kvantna prepletenost omogočata novo paradigmo procesiranja, ki obljublja eksponentno pospešitev nekaterih algoritmov glede na klasične računalnike. Izgradnja skalabilnega kvantnega računalnika trenutno otežuje zahteven nadzor neizoliranega sistema kvantnih bitov. Vendar predstavljene implementacije, množica razvitih kvantnih algoritmov in protokolov, dostopni računalniški simulatorji in možnost uporabe manjših fizičnih kvantnih računalnikov preko spleta dopuščajo preučevanje in razvoj področja.

Tako QCA kot kvantnemu procesiranju je skupna reverzibilnost. Fizično reverzibilno procesiranje brez izgube informacij se približa minimalni možni porabi energije. Kvantno procesiranje je inherentno reverzibilno, saj informacije v kvantnem bitu ni mogoče izbrisati. Primerna zasnova QCA vezja omogoča implementacijo fizično reverzibilne procesne platforme.

Navodila za izdelavo seminarske naloge

- Vaše datoteke se nahajajo v direktorijih `Skupina*`, kjer `*` predstavlja številko vaše skupine - glavna datoteka je `main.tex`.
- Slike shranjujte v svoj direktorij.
- Vse labelle začnite z znaki `g*`, kjer `*` predstavlja številko vaše skupine.
- Pri navajanju virov uporabite datoteko `references.bib`, ki se nahaja v ko-renskem direktoriju projekta.
- Pred dodajanjem novih virov v datoteko `references.bib` dobro preverite, če je vir mogoče že vsebovan v datoteki - v tem primeru se sklicujte na obstoječ vnos.

V okviru prvega seminarja se boste ukvarjali s procesnimi platformami, ki za delovanje izkoriščajo kvantne pojave. Delo bo potekalo v skupinah s štirimi študenti. Za skupinsko delo uporabljajte repozitorij, kot je npr. `dropbox`¹ ali `git`². Cilj seminarja so izdelki in poročila, ki jih lahko ob zaključku združimo v celoto (glej rezultate seminarjev iz prejšnjih let). Poročilo pišite v okolju LaTeX, kjer lahko za lažje skupinsko delo uporabljate okolje, kot je npr. `overleaf`³. Za iskanje virov uporabljajte iskalnike znanstvene literature⁴. Rok za izdelavo prvega seminarja je 24. 11. 2024. Predstavitve nalog bodo v tednu od 25. 11. 2024 v terminu vaj. Na zagovoru seminarja bo imela vsaka skupina 10 minutno predstavitev svojega izdelka, nato bo sledila krajša diskusija.

Za predlogo uporabite strukturo znanstvenega članka, ki obsega poglavja Uvod, Metode, Rezultati, Zaključek in Literatura. Slike naj bodo v formatu PDF ali EPS, z ustreznimi viri (literaturo) polnite vašo BIB datoteko. Poročilo redno izpopolnjujte. Ker gre za skupinsko delo, mora biti na koncu poročila tudi poglavje z naslovom Doprinosi avtorjev, kjer v enem stavku zapišite, kakšen je bil doprinos posameznega člana skupine.

¹ <https://www.dropbox.com>

² <https://bitbucket.org>

³ <https://www.overleaf.com/>

⁴ <https://scholar.google.si>, www.sciencedirect.com, <https://www.scopus.com>, <https://arxiv.org>, <http://citeseerx.ist.psu.edu>

Kvantni celični avtomati

Ena izmed tem za prvi seminar je področje kvantnih celičnih avtomatov (angl. QCA) [1, 2, 3, 4], ki so možna alternativna tehnologija procesiranja v prihodnosti. Cilj seminarja je lahko na primer zasnova in realizacija reverzibilne QCA strukture v okolju QCADesigner. Reverzibilno procesiranje je opisano v [5, 6, 7]. Primera osnovnih univerzalnih reverzibilnih logičnih operatorjev sta Toffolijeva in Fredkinova vrata. V seminarju izdelajte kompleksnejšo reverzibilno strukturo, na primer polni seštevalnik, LFSR (pomikalni register z linearno povratno vezavo) itd. Potreben pogoj za fizično reverzibilnost je logična reverzibilnost. Zato najprej zasnujte logično reverzibilno QCA strukturo. Logična reverzibilnost pomeni, da lahko iz logičnih vrednosti izhodov enolično določimo logične vrednosti vhodov, torej med vhodi in izhodi obstaja bijektivna preslikava. Nato poskusite z orodjem QCADesigner zasnovati še fizično reverzibilno strukturo QCA. Slednja mora biti najprej logično reverzibilna. Poleg tega mora izpolnjevati tudi dodatni pogoj: če spremenimo izhodne celice v vhodne in obratno ter določimo vhodne logične vrednosti (na prej izhodnih celicah), moramo na izhodi (prej vhodne celice) dobiti ustrezne vrednosti, določene z bijekcijo. Z reverzibilno strukturo lahko torej izračunamo vrednost bijektivne funkcije (npr. pretvorbo v dvojiški komplement, izračun zgoščevalnih (hash) funkcij itd.) in z zamenjavo vhodov in izhodov tudi njeno inverzno vrednost.

V poglavju Metode opišite, ali ste uporabili ad hoc metodo ali ste formalizirali metodo snovanja QCA strukture. Navedite tudi, ali prosto določite urino cono vsaki posamezni QCA celici ali je vaša struktura zasnovana z uporabo strukturiranih pravokotnih urin con.

QCA strukturo izdelajte s pomočjo orodja QCADesigner, s katerim tudi simulirajte in testirajte njeno delovanje. Poročilo izdelajte v okolju LaTeX in dopolnjujte repozitorij s pripadajočimi datotekami (QCAD datoteke, slike itd.).

Kvantni računalniki

Danes v računalništvu prevladujoča tehnologija CMOS se z miniaturizacijo tranzistorjev bliža vrhuncu učinkovitosti, ko je zaradi fizikalnih omejitev ne bo več možno izboljševati. Za nadaljnji napredek strojne računalniške opreme bo nujno prevzeti katero od alternativnih procesnih platform. Ena od teh je kvantno računalništvo, ki za procesiranje izkorišča kvantne pojave in operira z delci na nanometrskem nivoju. Poleg tega teoretične raziskave in tudi praktične implementacije prikazujejo večjo zmogljivost kvantnega računalništva od današnje tehnologije.

Kvantno računalništvo se s hitrim razvojem kaže kot perspektivna alternativa za dopolnitev in nadomestitev tehnologije CMOS. Številna velika podjetja so razvila in izdelala svoje kvantne računalnike ter prikazala njihovo delovanje. Med najbolj znanimi so izvedbe:

- IBM,
- Google,
- Intel,
- Rigetti,
- D-Wave Systems
- ter mnoga druga podjetja in univerze po celem svetu.

Raziskovalci so predstavili kvantne algoritme (npr. Shorov in Groverjev), ki imajo manjšo časovno kompleksnost kot najboljši klasični algoritmi. To bi lahko privedlo do eksponentno hitrejšega reševanja določenih problemov, vendar zaenkrat praktične implementacije teh algoritmov še niso skalabilne. V zadnjem času pa so fizično izdelali kvantne računalnike, ki nekatere probleme rešijo hitreje od najzmogljivejših klasičnih računalnikov, s čimer so prikazali t.i. "quantum supremacy". Med najbolj odmevnimi izvedbami sta Googlov procesor Sycamore [8] in računalnik Zuchongzhi, razvit na univerzi USTC [9].

Cilj seminarske naloge s področja kvantnega računalništva je spoznavanje osnovnih konceptov te procesne platforme. V nadaljevanju je naštetih nekaj predlogov tem (odebeljeno besedilo). Za seminarsko nalogo si izberite eno izmed njih, lahko pa predlagate tudi kakšno drugo temo s tega področja.

Ker kvantni računalniki še niso v splošni uporabi, se za analizo delovanja uporabljajo razni simulatorji, s katerimi lahko načrtujemo, implementiramo in simuliramo delovanje kvantnih algoritmov. V številnih programskih jezikih je bilo razvitih mnogo simulatorjev z različnimi funkcijami in lastnostmi. Obsežen seznam se nahaja na spletni strani Quantiki. Ena izmed tem seminarske naloge je **načrtovanje in implementacija kvantnih algoritmov v več različnih simulatorjih**. Pri tem predstavite prednosti in slabosti izbranih simulatorjev in njihove koristne funkcije.

Za fizično implementacijo kvantnih računalnikov so raziskovalci ponudili številne tehnologije in v izbranih dejansko realizirali procesorje. Med tehnologijami so superprevodna vezja, ujeti ioni, kvantne pike, jedrska magnetna resonanca, uporaba fotonov, elektronov in številni drugi predlogi. V seminarski nalogi lahko **predstavite predlagane tehnologije, njihove prednosti in slabosti ter primere fizične implementacije**.

Razvitih je bilo mnogo kvantnih algoritmov, ki predstavljajo reševanje problemov učinkoviteje kot s klasičnimi računalniki. V seminarju lahko **izvedete podrobno analizo kompleksnega kvantnega algoritma**. Pri tem algoritem načrtujete in implementirate v simulatorju, analizirate njegovo kompleksnost, simulirate delovanje z različnimi vhodnimi podatki in predlagate možne optimizacije katerega od parametrov algoritma.

Sčasoma želimo splošno uporabljati univerzalne kvantne računalnike, s katerimi lahko izvedemo poljubno aplikacijo, kot sedaj to omogočajo klasična vezja. Trenutno pa realizacije kvantnih računalnikov prikazujejo svojo učinkovitost pri reševanju specifičnih nesplošnih problemov. **Navedite nekatere od teh problem-skih področij (npr. boson sampling), jih podrobno opišite in predstavite obstoječe ter potencialne rešitve.**

Za izdelavo skalabilnih kvantnih računalnikov je zelo pomembno popravljanje napak pri kvantnem procesiranju, ki se zaradi lastnosti kvantnih delcev precej razlikuje od popravljanja napak pri klasičnem procesiranju. **Podrobno preučite algoritme za popravljanje napak pri kvantnem procesiranju, jih implementirajte, prikažite njihovo delovanje z uporabo simulatorja in analizirajte njihove lastnosti.**

Kazalo

1	Redukcija števila logičnih operatorjev za QCA	11
	Iztok Bajcar, Ana Luetić, Timotej Zgonik	
1.1	Uvod	11
1.1.1	Logične funkcije treh spremenljivk	12
1.1.2	QCA negator	14
1.1.3	QCA majoritetna vrata	14
1.1.4	Polni seštevalnik	16
1.2	Metode	18
1.3	Rezultati	20
1.4	Zaključek	20
1.5	Doprinosi avtorjev	21
	Literatura	23

Poglavje 1

Redukcija števila logičnih operatorjev za QCA

Iztok Bajcar, Ana Luetić, Timotej Zgonik

Povzetek Zaradi prostorskih in finančnih razlogov pri razvoju digitalnih vezij želimo za posamezne gradnike vezja porabiti čim manj prostora. Pri tem si pomagamo s poenostavljanjem in minimizacijo preklonnih funkcij, ki opisujejo vezje. Isto velja za vezja, realizirana s kvantnimi celičnimi avtomati (QCA). Eden od glavnih gradnikov vezij kvantnih celičnih avtomatov so majoritetna vrata, katerih implementacija zasede relativno veliko prostora. V nalogi z uporabo majoritetnih vrat realiziramo funkcije treh spremenljivk, pri čemer vezja predhodno poenostavimo.

1.1 Uvod

V tej seminarski nalogi analiziramo pristop poenostavitve funkcij treh spremenljivk iz članka *A Method of Majority Logic Reduction for Quantum Cellular Automata* [10] ter ponovimo in komentiramo izvedene eksperimente.

Iztok Bajcar
Fakulteta za računalništvo in informatiko Univerze v Ljubljani, e-mail: `ib8548@student.uni-lj.si`

Ana Luetić
Fakulteta za računalništvo in informatiko Univerze v Ljubljani, e-mail: `a196152@student.uni-lj.si`

Timotej Zgonik
Fakulteta za računalništvo in informatiko Univerze v Ljubljani, e-mail: `tz36793@student.uni-lj.si`

1.1.1 Logične funkcije treh spremenljivk

S tremi logičnimi spremenljivkami a , b in c lahko sestavimo 8 različnih mintermov, kot je razvidno iz tabele 1.1, vsako funkcijo treh spremenljivk $f(a, b, c)$ pa lahko zapišemo kot kombinacijo teh mintermov [10].

a	b	c	$minterm$
0	0	0	$\bar{a}\bar{b}\bar{c}$
0	0	1	$\bar{a}\bar{b}c$
0	1	0	$\bar{a}b\bar{c}$
0	1	1	$\bar{a}bc$
1	0	0	$a\bar{b}\bar{c}$
1	0	1	$a\bar{b}c$
1	1	0	$ab\bar{c}$
1	1	1	abc

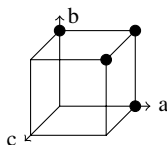
Tabela 1.1: Vsi možni mintermi, sestavljeni iz treh spremenljivk a , b in c .

Resničnostno tabelo ene od takšnih funkcij podajamo v tabeli 1.2. Če jo zapišemo v disjunktivni normalni obliki, dobimo $f(a, b, c) = \bar{a}b\bar{c} \vee a\bar{b}\bar{c} \vee ab\bar{c} \vee abc$.

a	b	c	$f(a, b, c)$
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

Tabela 1.2: Primer funkcije treh spremenljivk.

V 3D prostoru lahko vsaki osi priredimo eno od spremenljivk a , b in c . Minterme, ki nastopajo v formuli, potem lahko predstavimo kot oglišča enotske kocke (slika 1.1).



Slika 1.1: Enotska kocka, ki predstavlja funkcijo iz tabele 1.2, z ustrezno označenimi oglišči.

To nam omogoči poenostavitev formule tako, da najdemo najkrajšo kombinacijo ploskev, robov in oglišč kocke, ki skupaj pokrijejo vsa označena oglišča. V omenjenem primeru optimalno pokritje predstavljajo robovi ab , $b\bar{c}$ in $a\bar{c}$, s čimer dobimo formulo $f(a, b, c) = ab \vee b\bar{c} \vee a\bar{c}$ [10].

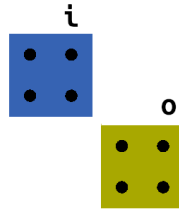
Na podlagi principa take enotske kocke članek predlaga uporabo 13 standardnih funkcij, ki omogočajo izvedbo vseh mogočih funkcij treh spremenljivk (slika 1.2). Te funkcije sicer same po sebi niso reverzibilne [10].

Number of points	3-Cube Structure	Description	Function No.	Standard Function
One point		one point	1	$F = ABC$
Two points		two points: one edge	2	$F = AB$
		two points: nonadjacent, but in one face	3	$F = ABC + \bar{A}\bar{B}\bar{C}$
		two points: nonadjacent, and not in one face	4	$F = ABC + \bar{A}\bar{B}\bar{C}$
Three points		three points: two edges sharing one common point	5	$F = AB + BC$
		three points: one edge and one nonadjacent point	6	$F = AB + \bar{A}\bar{B}\bar{C}$
		three points: all nonadjacent	7	$F = ABC + \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}\bar{C}$
Four points		four points: one face	8	$F = A$
		four points: three edges share one common point	9	$F = AB + BC + AC$
		four points: two edges perpendicular to each other	10	$F = AB + \bar{B}\bar{C}$
		four points: two edges sharing one common point and one nonadjacent point	11	$F = AB + BC + \bar{A}\bar{B}\bar{C}$
		four points: two parallel edges but not in a face	12	$F = AB + \bar{A}\bar{B}$
		four points: all nonadjacent	13	$F = ABC + \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}\bar{C}$

Slika 1.2: Shema 13 funkcij, ki pokrivajo vse mogoče funkcije treh spremenljivk.

1.1.2 QCA negator

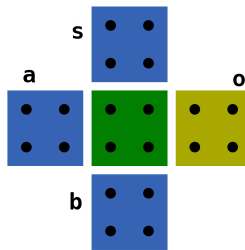
Negator je gradnik logičnega vezja, ki obrne svojo vhodno vrednost – preklopna funkcija, ki mu pripada, je enaka logični negaciji. V kontekstu kvantnih celičnih avtomatov ga lahko realiziramo tako, da dve QCA celici staknemo diagonalno, kot prikazuje slika 1.3.



Slika 1.3: Shema QCA negatorja. i označuje vhodno, o pa izhodno celico.

1.1.3 QCA majoritetna vrata

Majoritetna vrata so eden glavnih logičnih gradnikov v vezjih kvantnih celičnih avtomatov. Sprejmejo tri vhodne spremenljivke in na izhod prenesejo vrednost, ki se med njimi najpogosteje pojavi. Na sliki 1.4 je prikazana shema majoritetnih vrat, realiziranih s QCA, v tabeli 1.3 pa podajamo njihovo pravilnostno tabelo. Iz slednje je razvidna značilnost majoritetnih vrat, da lahko odvisno od vrednosti spremenljivke s delujejo bodisi kot logični *in* (ko velja $s = 0$) bodisi kot logični *ali* (ko velja $s = 1$), kar pomeni, da z njimi lahko realiziramo tako konjunkcijo kot tudi disjunkcijo dveh logičnih spremenljivk.



Slika 1.4: Shema majoritetnih vrat. Spremenljivke s , a in b so vhodne spremenljivke, o pa je izhodna spremenljivka.

s	a	b	o
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Tabela 1.3: Pravilnostna tabela majoritetnih vrat s slike 1.4.

Če privzamemo funkcijo $M(a, b, c)$, ki zavzame najpogostejšo vrednost izmed spremenljivk a , b in c , lahko zgornja majoritetna vrata zapišemo z logično funkcijo $o = M(s, a, b)$. Upoštevajoč lastnost majoritetnih vrat in posledično tudi funkcije M , da lahko izvajajo konjunkcijo ali disjunkcijo dveh vhodnih spremenljivk, lahko poenostavimo funkcije iz tabele 1.2 na naslednji način [10]:

$$f_1(a, b, c) = abc = M(M(a, b, 0), c, 0)$$

$$f_2(a, b, c) = ab = M(a, b, 0)$$

$$\begin{aligned} f_3(a, b, c) &= abc \vee a\bar{b}\bar{c} = \\ &= a(bc \vee \bar{b}\bar{c}) = \\ &= M(a, M(M(b, c, 0), M(\bar{b}, \bar{c}, 0), 1), 0) = \\ &= M(M(a, b, \bar{c}), M(a, \bar{b}, c), 0) \end{aligned}$$

$$\begin{aligned} f_4(a, b, c) &= abc \vee a\bar{b}\bar{c} = \\ &= M(M(a, b, 0), \bar{c}, 0) \vee M(M(\bar{a}, \bar{b}, 0), \bar{c}, 0) = \\ &= M(M(M(a, b, 0), c, 0), M(M(\bar{a}, \bar{b}, 0), c, 0), 1) \end{aligned}$$

$$\begin{aligned} f_5(a, b, c) &= ab \vee bc = \\ &= ba \vee bc = \\ &= b(a \vee c) = \\ &= M(b, M(a, c, 1), 0) \end{aligned}$$

$$\begin{aligned} f_6(a, b, c) &= ab \vee a\bar{b}c = \\ &= M(a, b, 0) \vee M(M(\bar{a}, \bar{b}), c, 0) = \\ &= M(M(a, b, 0), M(M(\bar{a}, \bar{b}), c, 0), 1) \end{aligned}$$

$$\begin{aligned}
f_7(a, b, c) &= abc \vee \bar{a}b\bar{c} \vee a\bar{b}\bar{c} = \\
&= b(ac \vee \bar{a}\bar{c}) \vee a\bar{b}\bar{c} = \\
&= M(M(b, M(M(a, c, 0), M(\bar{a}, \bar{c}, 0), 1), 0), M(M(a, \bar{b}, 0), c, 0), 1)
\end{aligned}$$

$$f_8(a, b, c) = a = M(a, 0, 1)$$

$$\begin{aligned}
f_9(a, b, c) &= ab \vee bc \vee ac = \\
&= M(a, b, c)
\end{aligned}$$

Opomba: ab , bc in ac so vse možne kombinacije dveh spremenljivk izmed podanih, izraz $ab \vee bc \vee ac$ pa je posledično resničen, če sta vsaj dve spremenljivki resnični, in neresničen v ostalih primerih, kar predstavlja prav funkcijo M .

$$\begin{aligned}
f_{10}(a, b, c) &= ab \vee \bar{b}c = \\
&= M(M(a, b, 0), M(\bar{b}, c, 0), 1)
\end{aligned}$$

$$\begin{aligned}
f_{11}(a, b, c) &= ab \vee bc \vee \bar{a}\bar{b}\bar{c} = \\
&= b(a \vee c) \vee \bar{a}\bar{b}\bar{c} = \\
&= M(M(b, M(a, c, 1), 0), M(M(\bar{a}, \bar{b}, 0), \bar{c}, 0), 1)
\end{aligned}$$

$$\begin{aligned}
f_{12}(a, b, c) &= ab \vee \bar{a}\bar{b} = \\
&= M(a, b, 0) \vee M(\bar{a}, \bar{b}, 0) = \\
&= M(M(a, b, 0), M(\bar{a}, \bar{b}, 0), 1)
\end{aligned}$$

$$\begin{aligned}
f_{13}(a, b, c) &= abc \vee \bar{a}\bar{b}c \vee a\bar{b}\bar{c} \vee \bar{a}b\bar{c} = \\
&= a(bc \vee \bar{b}\bar{c}) \vee \bar{a}(\bar{b}c \vee b\bar{c}) = \\
&= M(M(a, M(M(b, c, 0), M(\bar{b}, \bar{c}, 0), 1), 0), \\
&\quad M(\bar{a}, M(M(\bar{b}, c, 0), M(b, \bar{c}, 0), 1), 0), 1) = \\
&= M(M(\bar{a}, b, c), M(a, \bar{b}, c), \bar{c})
\end{aligned}$$

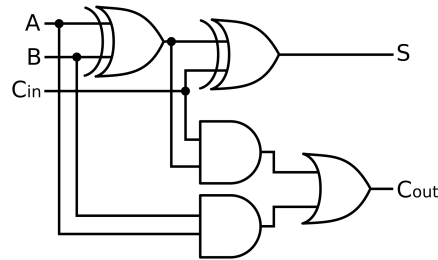
1.1.4 Polni seštevalnik

Polni seštevalnik je logično vezje, ki omogoča seštevanje dvojiških števil a in b , pri čemer upošteva prenos (carry) iz predhodne stopnje (c_{in}), na izhodu pa dobimo 1-bitno vsoto seštevancev s in izhodni prenos c_{out} . Tabela 1.4 predstavlja pripadajočo pravilnostno tabelo.

a	b	c_{in}	c_{out}	sum
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

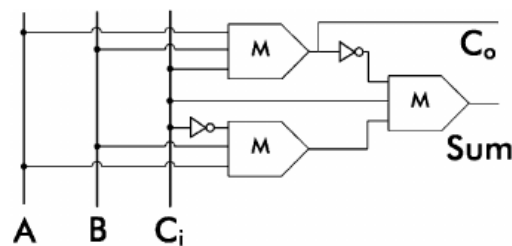
Tabela 1.4: Pravilnostna tabela polnega seštevalnika.

Na sliki 1.5 je prikazana logična shema enobitnega polnega seštevalnika iz logičnih vrat AND, OR in XOR.



Slika 1.5: Logična shema polnega seštevalnika iz vrat AND, OR in XOR [11].

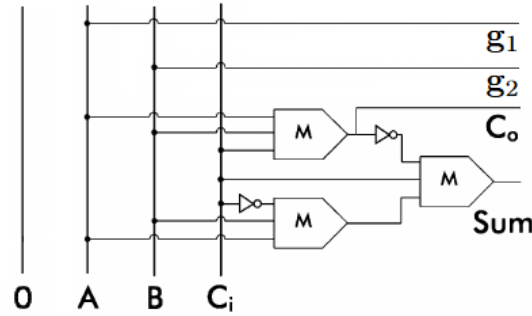
Ker lahko vrata AND in OR implementiramo z majoritetnimi vrati in ker lahko $a \text{ XOR } b$ implementiramo kot $(a \vee b) \wedge \neg(ab)$, lahko posledično celoten seštevalnik implementiramo le z uporabo majoritetnih vrat in negatorjev, kot prikazuje slika 1.6. V nadaljevanju naloge bomo to realizacijo poustvarili in simulirali.



Slika 1.6: Logična shema polnega seštevalnika, realiziranega z uporabo majoritetnih vrat in negatorjev [10].

Klasičen polni seštevalnik ni reverzibilen. Pogoji reverzibilnosti je enoličnost izhodnih vrednosti. Ker se število vhodnih podatkov polnega seštevalnika razlikuje od števila izhodnih podatkov, ta pogoj ni izpolnjen. Poleg tega sta izhoda $(s, c_{out}) = (1, 0)$ in $(s, c_{out}) = (0, 1)$ oba rezultata treh različnih kombinacij vhodnih podatkov. Zato dodajanje ene izhodne spremenljivke ne zadostuje za reverzibilnost.

Polni seštevalnik transformiramo v reverzibilen logični operator s štirimi vhodnimi in štirimi izhodnimi podatki tako, da dodamo vhodni podatek s fiksno vrednostjo 0 ter dva izhodna podatka $g_1 = A$ in $g_2 = B$. Logična shema tega operatorja je prikazana na sliki 1.7.



Slika 1.7: Logična shema reverzibilnega polnega seštevalnika.

Njegov inverz lahko potem poiščemo s pomočjo naslednjih formul.

$$A = g_1$$

$$B = g_2$$

$$C_i = (S \wedge \overline{C_{out}} \wedge \bar{A} \wedge \bar{B})$$

$$\vee (\bar{S} \wedge C_{out} \wedge \bar{A} \wedge B)$$

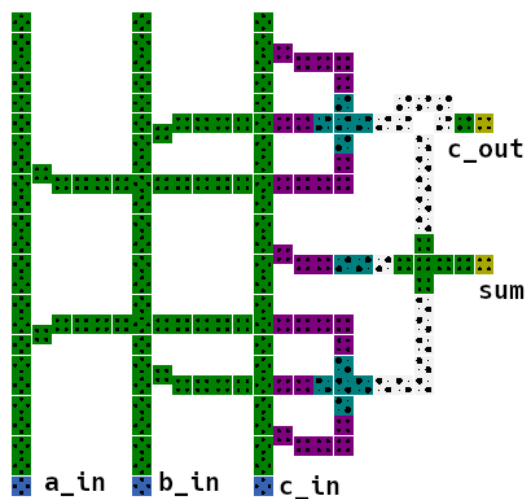
$$\vee (\bar{S} \wedge C_{out} \wedge A \wedge \bar{B})$$

$$\vee (S \wedge C_{out} \wedge A \wedge B)$$

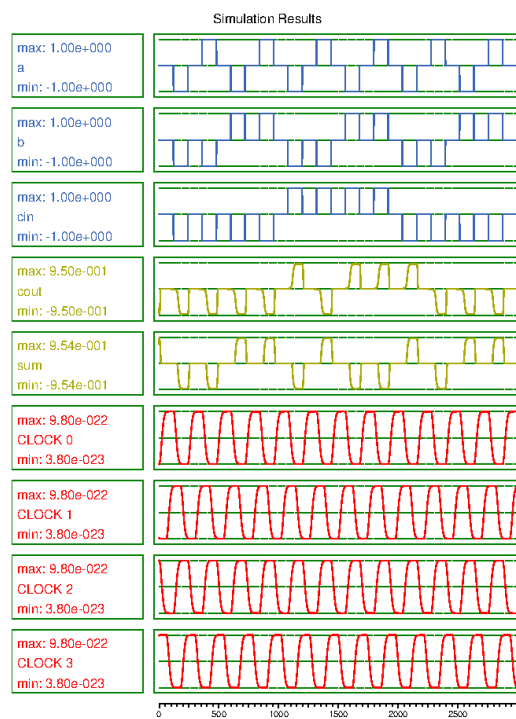
1.2 Metode

Za načrtovanje vezja smo uporabili program QCADesignerE. Končno vezje prikazuje slika 1.8.

Vezje smo tudi simulirali in s tem preverili pravilnost delovanja (slika 1.9). Na sliki vidimo, da vrednosti izhodnih celic glede na vrednosti vhodnih celic sovpadajo z resničnostno tabelo polnega seštevalnika (1.4).

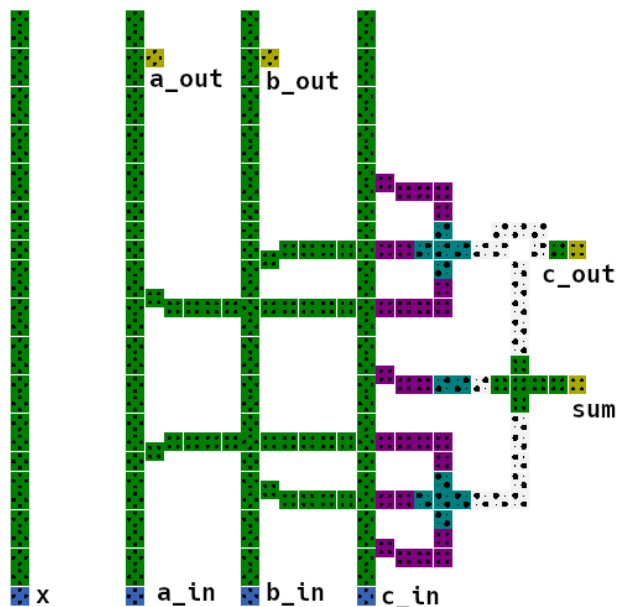


Slika 1.8: Posnetek sheme našega polnega seštevalnika, simuliranega s programom QCADesignerE.



Slika 1.9: Rezultat simulacije polnega seštevalnika.

Simulirali smo tudi reverzibilen logični operator s sheme 1.7. Vezje je prikazano na sliki 1.10.



Slika 1.10: Posnetek sheme našega reverzibilnega polnega seštevalnika, simuliranega v QCADesignerju.

1.3 Rezultati

Ugotavljamo, da predlagana metoda poenostavitve vezij deluje, z njeno pomočjo pa smo uspešno realizirali polni seštevalnik z uporabo minimalnega števila majoritetnih vrat. Prav tako smo uspeli načrtovati teoretično reverzibilen polni seštevalnik, ki bi po podatkih programa QCADesigner, če bi ga fizično realizirali s kvantnimi celičnimi avtomati, zasedal površino 0,39 kvadratnih mikrometrov, kar je približno 7-krat večje od originalne implementacije [10].

1.4 Zaključek

V okviru seminarske naloge smo utemeljili, da lahko vsako funkcijo treh spremenljivk zapišemo kot kombinacijo 13 standardnih funkcij. Te funkcije smo nato preo-

blikovali tako, da v formulah nastopajo majoritetna vrata kot edini logični operator. Na ta način lahko za posamezne funkcije porabimo manj prostora.

Kot primer funkcije treh spremenljivk smo obravnavali polni seštevalnik. Utemeljili smo, da lahko logično shemo, sestavljeno iz logičnih vrat AND, OR in XOR, preoblikujemo v logično shemo, sestavljeno le iz logičnih majoritetnih vrat in negatorjev. Ker tak operator ni reverzibilen, smo ga nato dopolnili do reverzibilnega.

Vezje obeh operatorjev smo zasnovali s programom QCADesignerE. Simulirali smo delovanje in se tako prepričali o pravilnosti zasnove.

1.5 Doprinosi avtorjev

Iztok Bajcar je opisal, kako lahko funkcije treh spremenljivk predstavimo s 13 standardnimi funkcijami, in sodeloval pri implementaciji in simulaciji operatorjev.

Ana Luetić je opisala, kako lahko polni seštevalnik razširimo do reverzibilnega, in sodelovala pri implementaciji in simulaciji operatorjev.

Timotej Zgonik je opisal, kako polni seštevalnik sestavimo z uporabo samo majoritetnih vrat in negatorjev, in sodeloval pri implementaciji in simulaciji operatorjev.

Literatura

1. C. S. Lent, P. D. Tougaw, W. Porod, and G. H. Bernstein, "Quantum cellular automata," *Nanotechnology*, vol. 4, no. 1, p. 49, 1993.
2. P. D. Tougaw and C. S. Lent, "Logical devices implemented using quantum cellular automata," *Journal of Applied Physics*, vol. 75, no. 3, pp. 1818–1825, 1994.
3. C. S. Lent and P. D. Tougaw, "A device architecture for computing with quantum dots," *Proceedings of the IEEE*, vol. 85, no. 4, pp. 541–557, 1997.
4. R. Cowburn and M. Welland, "Room temperature magnetic quantum cellular automata," *Science*, vol. 287, no. 5457, pp. 1466–1468, 2000.
5. T. Toffoli, "Reversible computing," in *Proceedings of the 7th Colloquium on Automata, Languages and Programming*, (London, UK, UK), pp. 632–644, Springer-Verlag, 1980.
6. "The fantastical promise of reversible computing," <https://www.technologyreview.com/s/422511/the-fantastical-promise-of-reversible-computing>, 2011.
7. M. P. Frank, "Throwing computing into reverse," *IEEE Spectrum*, vol. 54, no. 9, pp. 32–37, 2017.
8. F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, S. Boixo, F. G. Brandao, D. A. Buell, *et al.*, "Quantum supremacy using a programmable superconducting processor," *Nature*, vol. 574, no. 7779, pp. 505–510, 2019.
9. Y. Wu, W.-S. Bao, S. Cao, F. Chen, M.-C. Chen, X. Chen, T.-H. Chung, H. Deng, Y. Du, D. Fan, *et al.*, "Strong quantum computational advantage using a superconducting quantum processor," *arXiv preprint arXiv:2106.14734*, 2021.
10. R. Zhang, K. Walus, W. Wang, and G. Jullien, "A method of majority logic reduction for quantum cellular automata," *Nanotechnology, IEEE Transactions on*, vol. 3, pp. 443 – 450, 01 2005.
11. C. M. Burnett, "Full adder," 2006.