

Univerza v Ljubljani



Univerza v Ljubljani
Fakulteta za Računalništvo in Informatiko
Tržaška 25, Ljubljana

N-bitni seštevalnik

Poročilo pri predmetu Optične in nanotehnologije

MENTOR: PRIMOŽ PEČAR
AVTORJI: JANI ČERNE, 63040020
JAN VESEL, 63050127
MARKO ŽIVEC, 63030187
JURE JANŠA, 63030118
JERNEJ SVETEK, 63030054

KAZALO

<u>1. Uvod.....</u>	<u>3</u>
<u>1.1. kvantni celični avtomat - QCA.....</u>	<u>3</u>
<u>1.2. QCA celica</u>	<u>3</u>
<u>1.3. QCA strukture</u>	<u>3</u>
<u>1.4. Ura</u>	<u>3</u>
<u>2. Naloga.....</u>	<u>5</u>
<u>2.1. opis problema.....</u>	<u>5</u>
<u>2.2. A+B+Cin seštevalnik.....</u>	<u>5</u>
<u>2.3 4-bitni polni seštevalnik.....</u>	<u>6</u>
<u>2.4. 2-bitni polni seštevalnik z možnostjo hold.....</u>	<u>8</u>
<u>3. Rezultati.....</u>	<u>10</u>
<u>3.1 Rezultati 4-bitne relaizacije.....</u>	<u>10</u>
<u>3.2 rezultati 2-bitnega seštevalnika z opcijo hold.....</u>	<u>12</u>
<u>4. Zaključek</u>	<u>13</u>
<u>5. Literatura.....</u>	<u>14</u>

1. UVOD

Pri predmetu ONT smo se spoznali z novimi tehnologijami, ki nam omogočajo nadaljnji razvoj in napredek na področju računalniške opreme in procesiranja. Ena izmed takih tehnologij je nanotehnologija. Naša naloga je realizirati n-bitni seštevalnik s QCA celicami. Mi smo si izbrali 4-bitni polni seštevalnik.

1.1. KVANTNI CELIČNI AVTOMAT - QCA

QCA je definiran in temelji na nanotehnologiji kot dvo- ali več- stanjski sistem. Osnovni element strukture je celica. Za kvantni celični avtomat je značilna zelo majhna poraba energije, hiter urin takt in je brez notranjih metaliziranih povezav. Njegov največji problem pa je občutljivost na vplive okolja, zaradi majhne razlike v energiji med obema stanjema celice ter tehnološki proces izdelave.

1.2. QCA CELICA

Dinamika v celicah temelji na zakonih kvantne fizike. Celico si lahko interpretiramo kot avtomat kvadratne oblike. QCA celica je sestavljena iz štirih polprevodniških pik, katere so povezane s štirimi tuneli, skozi katere lahko elektrona prehajata med kvantnimi pikami. Ker imata oba elektrona negativni naboj, se medsebojno odbijata, kar pomeni da poskušata biti med seboj kar se da narazen (Coulombov zakon). Razlikujemo dva tipa celic- osnovno in fiksno. V osnovni elektrona neprestano krožita v največji medsebojni oddaljenosti, v fiksni celicah pa sta fiksno polarizirana v enem od dveh stanj (»-1« – logična ničla ali »1« – logična enica). Ko pa postavimo fiksno celico poleg osnovne, se osnovna celica polarizira, torej se postavi v energetske minimalno stanje. Če se elektrona nahajata v diagonalah, pravimo, da je celica stabilna in da se nahaja v osnovnem stanju (logična ničla ali enica).

1.3. QCA STRUKTURE

Iz posameznih QCA celic lahko tvorimo kompleksnejše elemente. Z majoritetnimi vrati lahko tvorimo logično IN in ALI operacijo. Iz celic postavljenih v vrsto lahko tvorimo »žico«, če pa le te celice zavrtimo za 45° , pa lahko tvorimo alternirajočo linijo. Če dve liniji staknemo tako, da se stikata v vogalih, dobimo logično negacijo. Ena večjih prednosti QCA struktur je, da lahko brezizgubno križamo dve liniji. Za križanje potrebujemo eno navadno in eno za 45° stopinj zamaknjeno linijo. Iz teh preprostih osnovnih struktur je mogoče zgraditi kompleksnejše strukture, vendar se z večanjem števila celic začnejo večati tudi medsebojni vplivi, kar lahko privede do nenavadnih in nepredvidljivih situacij.

1.4. URA

Namen ure je izogibanje polarizacije v napačni smeri, omogoča sekvenčno obdelavo logičnih funkcij, prav tako pa služi za ojačevanje signala. Ura deluje v obliki električnega polja nad množicami celic. Ura nima neposredne povezave s celicami, deluje pa tako, da kontrolira zapiranje in odpiranje tunelov med pikami v posameznih kvantnih celicah. To pomeni, da imamo dejansko več ur s katerimi kontroliramo delovanje v celicah. Ura v QCA ima štiri faze: fazo preklopa, fazo zadrževanja, fazo sproščanja in fazo sproščenosti. V prvi fazi se ob dvigovanju pregrad celice pričnejo polarizirati, glede na vpliv sosednjih celic. V naslednji so pregrade dvignjene, tako da je tuneliranje onemogočeno, povedano drugače, stanje celic se več ne spreminja. V zadnjih dveh fazah se pregrade spustijo in elektroni postanejo prosti.

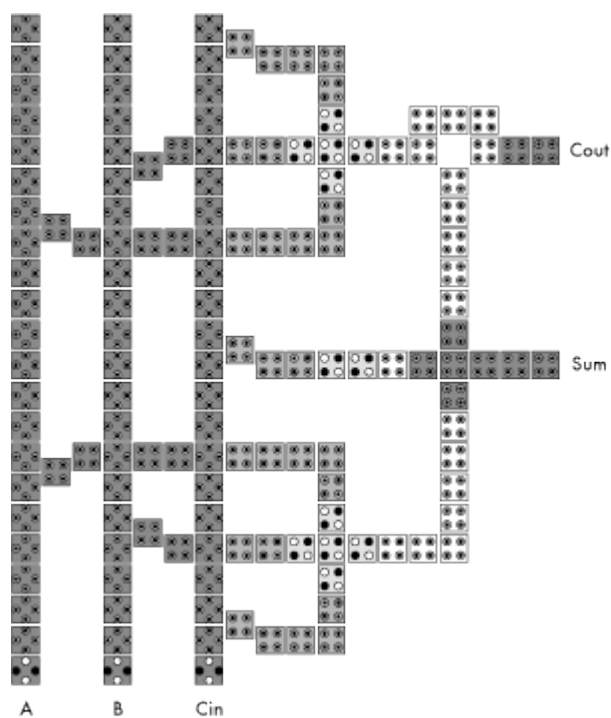
2. NALOGA

2.1. OPIS PROBLEMA

Realizirati smo morali n-bitni polni seštevalnik. Za realizacijo je bilo na voljo orodje QCADesigner. S pomočjo člankov, ki smo jih našli na internetu smo naredili 1-bitni polni seštevalnik, katerega smo nato razširili na 4-bite ter dodali še 4 registre za pomnjenje.

2.2. $A+B+C_{IN}$ SEŠTEVALNIK

Izhajali smo iz $A+B+C_{in}$ seštevalnika.

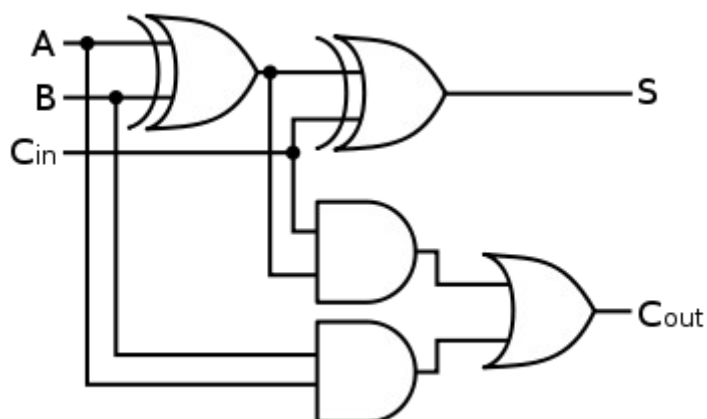


Slika predstavlja navedeni seštevalnik. Kaj pravzaprav seštevalnik je? Je digitalno vezje, ki sešteva dve števili. Ima tri vhode: števili A, B in prenos ($carry_{in}$) C_{in} ter 2 izhoda vsoto in prenos ($carry_{out}$) C_{out} .

Formula za vsoto (SUM): $(A \text{ xor } B) \text{ xor } C_{in}$

Formula za prenos (C_{out}): $(A \text{ and } B) \text{ or } (C_{in} \text{ and } B) \text{ or } (C_{in} \text{ and } A)$

Logična shema za polni seštevalnik s pravilnostno tabelo:



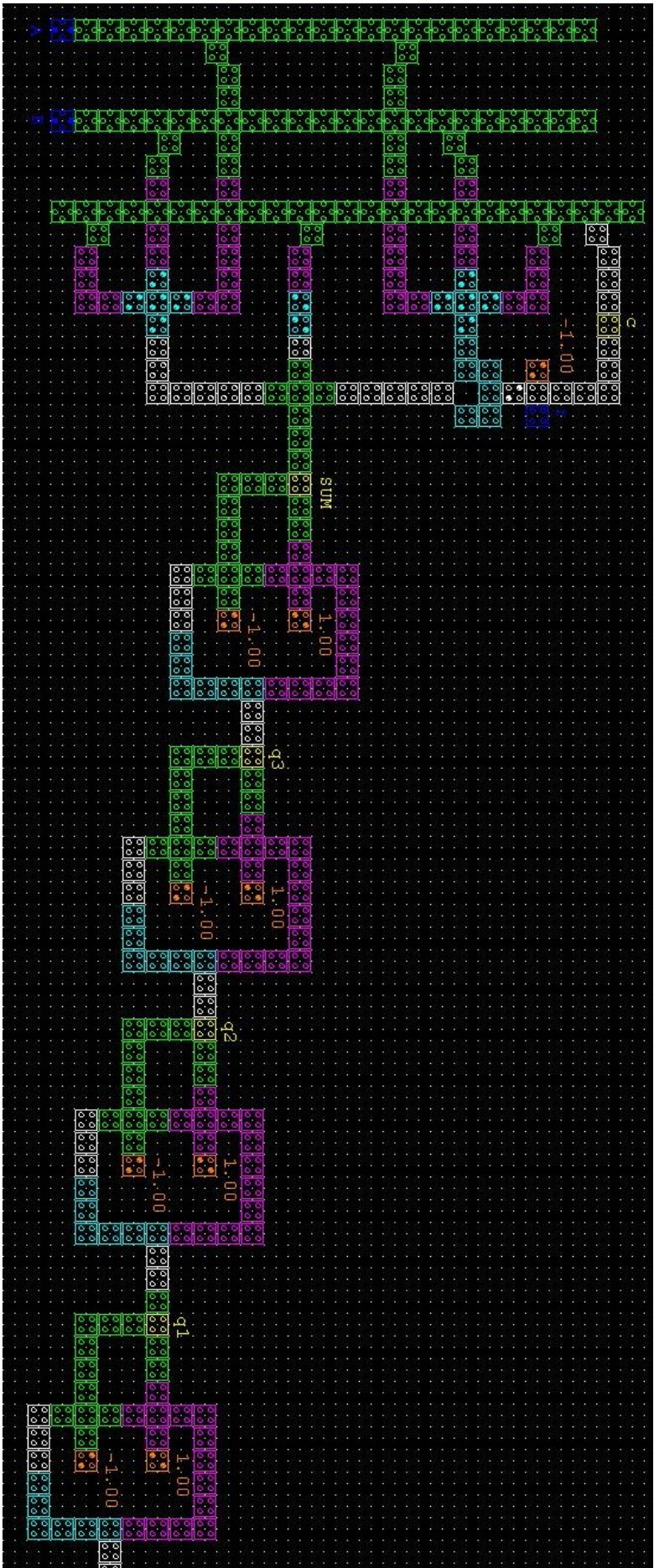
Input Output

<i>A</i>	<i>B</i>	<i>C_i</i>	<i>C_o</i>	<i>S</i>
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

Pravilnostna tabela

2.3 4-BITNI POLNI SEŠTEVALNIK

Naš seštevalnik deluje tako, da seštje števili A in B ter njima doda prenos (c bit). Rezultat se nato shrani v 4 RS pomnilne celice. Tako v njih hranimo 4 bitno število.

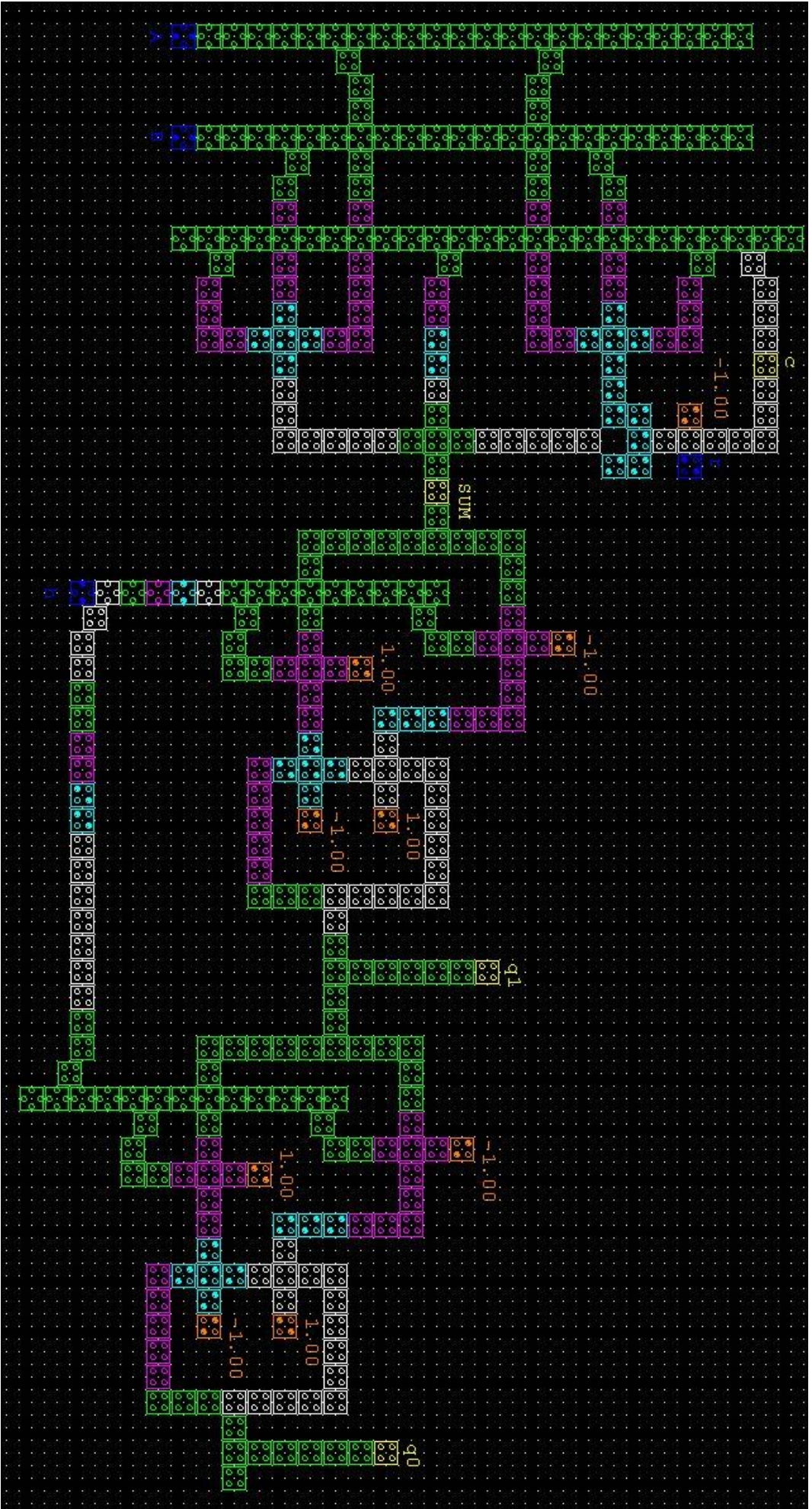


Iz začetne verzije, smo naredili seštevalnik, kateri upošteva prenos (carry bit) od prejšnjega koraka. Dodali smo še 4 RS celice, s katerimi smo naredili 4 bitni register. Po vsakem koraku, se vrednost rezultata (sum) shrani v prvo RS celico (Q4). Nato se zapis v vsakem naslednjem koraku prestavi eno nižje. Problem te realizacije je, da si ne zna zapomniti v določenem trenutku danih vrednosti. Če imamo preveliko število korakov, se nam bodo vrednosti iz zadnje celice prepisale z vrednostjo prejšnje.

2.4. 2-BITNI POLNI SEŠTEVALNIK Z MOŽNOSTJO HOLD

Na dani realizaciji imamo možnost ohraniti vrednost v določenem trenutku.

Z uporabo dodatnega signala H (hold), smo dobili naslednjo opcijo, da v trenutko ko ga postavimo na 1 nam ohranja vrednosti na izhodih Q1 in Q0. Z postavitvijo na 0, se nam ponovno zapomni vrednost SUM.



3. REZULTATI

3.1 REZULTATI 4-BITNE RELAIZACIJE

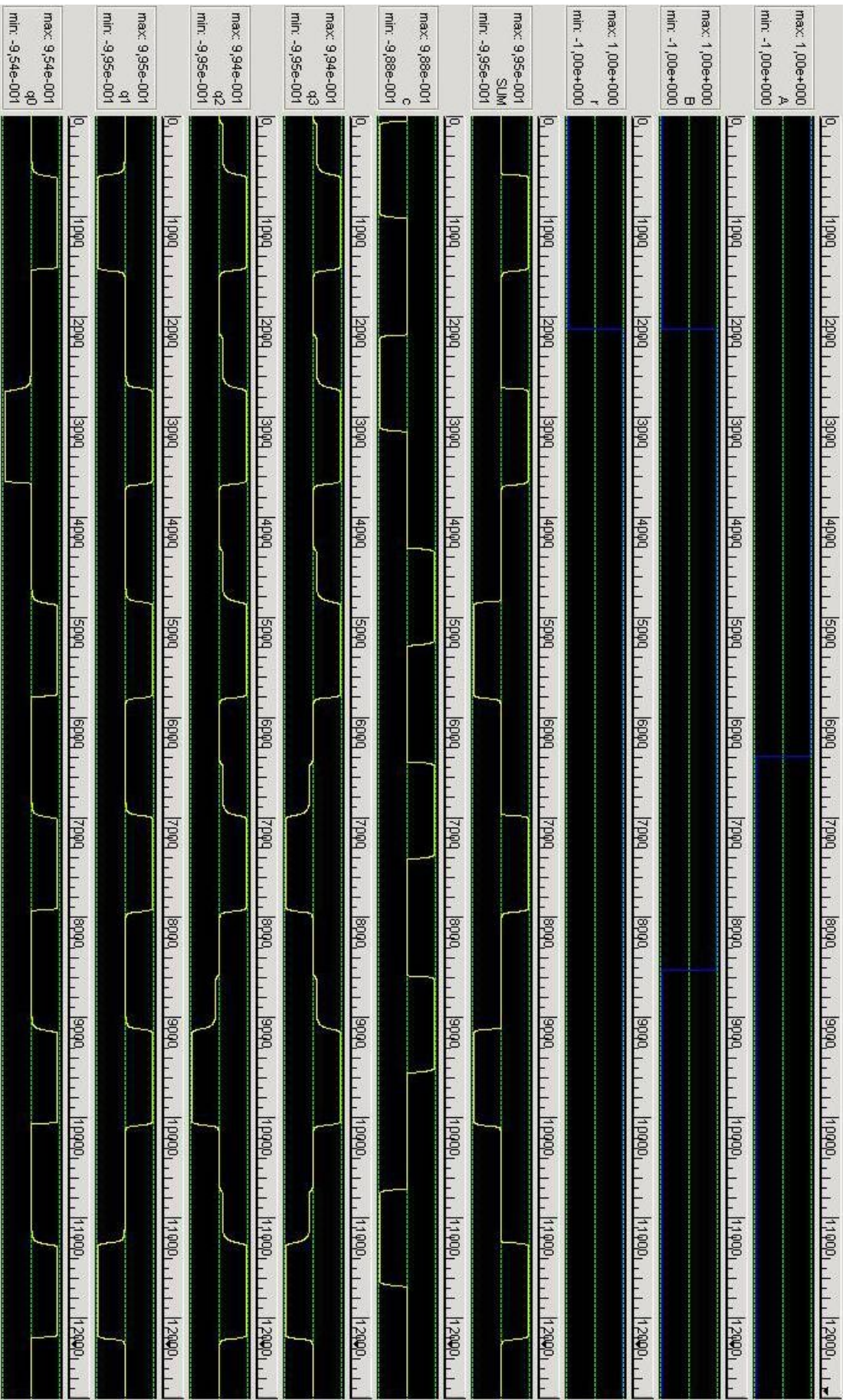
Pri testiranju našega seštevalnika smo uporabili testni vektor:

Inputs	Active	0	1	2	3	4	5
<input checked="" type="checkbox"/> A	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> B	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> r	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Lastnosti seštevalnika:

- realiziran je s 331 celicami
- časovni zamik (delay) = 5

Za rezultat smo dobili naslednjo sliko:



3.2 REZULTATI 2-BITNEGA SEŠTEVALNIKA Z OPCIJO HOLD.

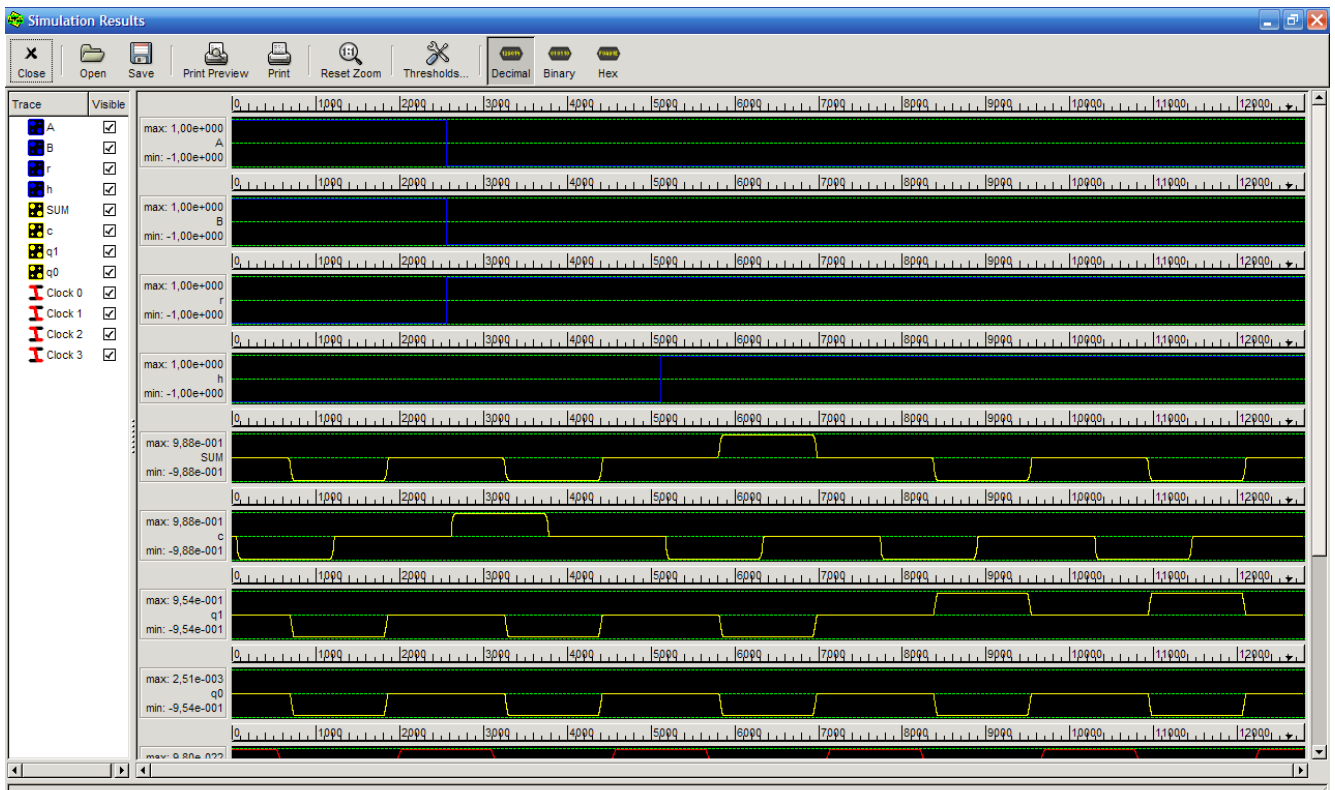
Pri testiranju našega seštevalnika smo uporabili testni vektor:

Inputs	Active	0	1	2	3	4
A	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
B	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
r	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
h	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Lastnosti seštevalnika:

- realiziran je s 375 celicami
- časovni zamik (delay) = 3

Za rezultat smo dobili naslednjo sliko:



4. ZAKLJUČEK

Pri reševanju problema so nam bili v veliko pomoč članki, ki smo jih poiskali na internetu. Zelo hitro smo odkrili tudi nekaj hroščev programa QCA designer v katerem smo programirali naš polni seštevalnik. Nalogo smo uspešno realizirali. Za primer smo naredili 4-bitni seštevalnik, ki pravilno deluje.

Uspelo nam je realizirati tudi 2-bitni seštevalnik z možnostjo "holda". Prva verzija nam je povzročila kar nekaj sivih las, saj se je animacija v QCA designerju pravilno izvedla vendar je bil rezultat simulacije napačen. Seveda smo večkratnim testiranjem in preurejanju QCA celic le uspeli najti delujočo rešitev.

Obe verziji seštevalnikov se, da razširiti na n-bitne seštevalnike z dodajanjem n-pomnilih celic.

5. LITERATURA

Zapiski vaj in predavanj.

Internetne strani: - <http://www.qcadesigner.ca/tutorials/QCATutorial.html>
- <http://www.ansijournals.com/jas/2007/3460-3468.pdf>
- <http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=04030918>