Dajana Danilović

Hardverska implementacija celularnih automata kao generatora slučajnih brojeva

Princip rada realizovanog pseudoslučajnog generatora, zasniva se na radu četvorosusednog celularnog automata čije su ćelije povezane asimetrično. Eksperimentalno je pronađeno nekoliko prenosnih funkcija i načina povezivanja ćelija pri kojima se automat ponaša kao dobar pseudoslučajan generator koji prolazi kriterijume DIEHARD baterije testova. Nakon simulacije uređaja u VHDL-u, generator je implementiran na FPGA čipu SPARTAN-3E. Prednost primene ovakvog generatora pseudoslučajnih brojeva u sistemima je velika brzina rada i mala potrošnja.

Uvod

Praktična upotreba generatora slučajnih brojeva raste sa razvojem tehnologije. Visoko kvalitetna, brzo generisana sekvenca slučajnih brojeva potrebna je u mnogim oblastima kao što su računarske simulacije, statistička i matrična izračunavanja, kriptografija, razne igre...

Postoje tri načina za dobijanje slučajnih brojeva: tablice, slučajni procesi u prirodi (npr. nivo šuma u elektronskoj napravi) i računarski generisani "pseudoslučajni" brojevi.

Različite primene zahtevaju nizove slučajnih brojeva veoma velike dužine, tako da njihova implementacija mora biti efikasna, brza i ne preterano složena. Postoji relativno mnogo metoda za stvaranje pseudoslučajnih brojeva, ali odabir neke od metoda treba da bude dobro prilagođen onome što se od generatora traži. Formule za generisanje brojeva mogu biti vrlo jednostavne, a da opet generisana sekvenca zadovolji parametre slučajnosti. Složeni algoritmi imaju prednost kada se želi što bolje sakriti način generisanja i tako otežati eventualno pogađanje formule kojom su brojevi dobijeni.

Celularni automati (CA)

Celularni automati nastali su krajem četrdesetih godina dvadesetog veka kao produkt ideje naučnika Džona Nojmana i Stanislava Ulmana. Osnovna ideja je bila u konstrukciji matematičkog modela koji simulira ponašanje sistema samoorganizacije, inače zastupljenih u prirodi. Poznavajući ovakav model, moguće je doneti značajne zaključke o osobinama i ponašanju sistema koji su zastupljeni u hemiji, fizici, biologiji... Znajući da, na osnovu drugog zakona termodinamike, izolovani sistemi, kao ovi koje simuliraju celularni automati, teže povećanju entropije dolazi se na ideju da CA mogu poslužiti i kao generatori slučajnih brojeva.

Celularni automat (Wolfram 1983) predstavlja dinamički sistem promenljiv u vremenskom i prostornom domenu. Čini ga mreža osnovnih jedinica – ćelija u n-dimenzionalnom prostoru. Mrežu karakteriše i međusobna povezanost osnovnih jedinica. *Konekcija* je način na koji su ćelije među sobom povezane. Ćelije koje su direktno povezane sa datom ćelijom jesu njeni *susedi*. Kod uniformnih automata susedi su definisani na isti način za sve ćelije u prostoru. Svaka ćelija u određenom vremenskom trenutku može imati jedno od konačnog broja stanja. Stanje svake ćelije u vremenu se menja u zavisnosti od stanja njenih suseda. Pravilo po kome se određuje stanje ćelije (u trenutku i) u zavisnosti od vrednosti stanja onih sa kojima je povezana (u trenutku i-1) naziva se *prenosna funkcija*.

Celularni automat implementiran na računaru, realizuje se tako što se pri svakom taktu menja konfiguracija CA. Način na koji se menja konfiguracija

Dajana Danilović (1987), Valjevo, ADRESA, učenica 4. razreda Valjevske gimnazije

MENTOR:
Đorđe Nijemčević, apsolvent Elektrotehničkog fakulteta u Beogradu

zavisi od karakteristika automata: konekcije i prenosne funkcije.

Karakteristike hardvera

Celularni automati se mogu simulirati na procesoru opšte namene. Zbog osobina koje poseduju, FPGA (Field Programmable Gate Array) čipovi su pogodniji za implementaciju CA.

FPGA čipovi sadrže veliki broj ćelija, multipleksera, logičkih kola, flip-flopova kao i mogućnost da se isti povežu u složenu celinu. To su programabilne logičke mreže koje se mogu konfigurisati tako da simuliraju bilo kakvu elektronsku shemu. Zahvaljujući stalnom povećanju gustine pakovanja uz istovremeni pad cene, danas se sve veći broj sistema na čipu realizuje u FPGA tehnologiji.

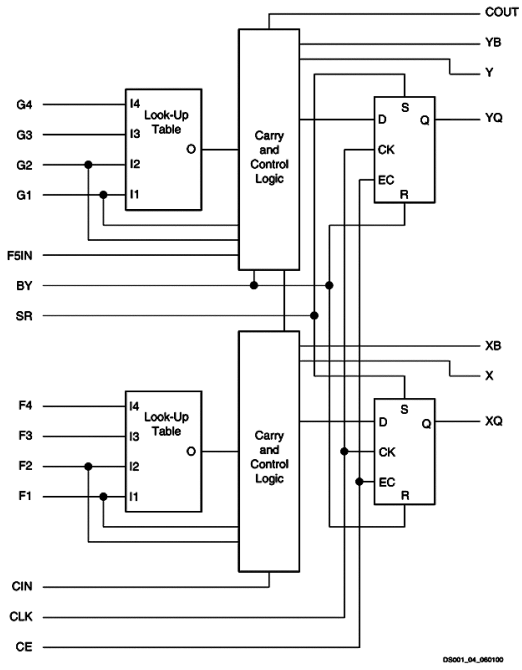
U konkretnom slučaju, planirano je da se implementacija generatora slučajnih brojeva izvrši na čipu Spartan 3E. Gradivni elementi čipa su:

- ulazno-izlazni blok IOB (Input/Output Block)
- programabilni logički blok CLB (Configurable Logic Block)
- blok RAM
- množači (Dedicated Multipliers)
- blok za digitalnu kontrolu takta DCM (Digital Clock Manager)

Mreža programabilnih logičkih blokova čini osnovni deo čipa. Shema CLB-a prikazana je na slici 1. U okviru CLB-a nalaze se logički generatori funkcija, memorijski element, multiplekseri, carry logika, i aritmetička kola. Ovi elementi se mogu različito povezati i podesiti u zavisnosti od namene bloka. Generator funkcija ili LUT (Look-Up Table) ima ulogu da formira bilo koju logičku funkciju sa četvorobitnim ulaznim podacima. Ovu njegovu karakteristiku bitno je uzeti u obzir pri konfigurisanju samog čipa radi optimalne iskorišćenosti hardverskih resursa.

Velika pogodnost realizacije uređaja na ovakvom čipu je mogućnost paralelnog izvršavanja procesa. Zapravo, pri svakom taktu, promena stanja svih ćelija se vrši istovremeno, ukoliko je čip tako konfigurisan. Mogućnost paralelnog izvršavanja predstavlja veliku prednost jer povećava brzinu uređaja.

Konfiguracija čipa se vrši pomoću koda napisanog u nekom od HDL (Hardware Description



Slika 1. Shematski prikaz CLB komponente čipa Spartan-3E

Figure 1.

Languages). Uređaj je opisan VHDL (Perry 2002) kodom koristeći pri tom softverski alat Xilinix-WebPack. VHDL (VHSIC hardware description language) je jezik za opisivanje hardvera, visokog nivoa koji se koristi za konfigurisanje elektronskih kola.

Prikazan je deo VHDL koda koji opisuje konfiguraciju osnovne jedinice CA-ćelije. Dat je kod ćelije za jednu od mogućih prenosnih funkcija. ćelija je definisana kao entitet, tj. modularna celina sa određenom strukturom i funkcijom. Definisani su ulazni i izlazni portovi kao obavezan deo opisa strukture. Struktura, kao i ponašanje entiteta su dalje objašnjeni u delu za opis njegove arhitekture. Konkretno, u ovom delu je opisana prenosna funkcija ćelije. Ulazni signali CLK, set i reset promenom vrednosti utiču direktno na stanje same ćelije. Njihov uticaj je definisan u okviru procesa (process (CLK, set, reset)). Na CLK ulaz dovodi se signal takta. Uzlazna ivica takta na CLK ulazu omogućava promenu stanja ćelije. Promena stanja ćelije zavisi od reset, set i clock_enable ulaznih signala. Signali

```
entity CELL is
generic     (     funkcija : STD_LOGIC_VECTOR(15 downto 0) :=
"1010101010101010");
      port(
            INPUT :           in    STD_LOGIC_VECTOR(3 downto 0);
            F     :           out   STD_LOGIC;
            CLK   :           in    STD_LOGIC;
            set, reset:       in    STD_LOGIC;
            clock_enable:     in    STD_LOGIC;
      );
end CELL;

architecture cell of CELL is
      signal F_ASYNC    :     STD_LOGIC;
begin
      F_ASYNC     <=    funkcija (0)      when  INPUT="0000" else
                        funkcija (1)      when  INPUT="0001" else
                        funkcija (2)      when  INPUT="0010" else
                        funkcija (3)      when  INPUT="0011" else
                        funkcija (4)      when  INPUT="0100" else
                        funkcija (5)      when  INPUT="0101" else
                        funkcija (6)      when  INPUT="0110" else
                        funkcija (7)      when  INPUT="0111" else
                        funkcija (8)      when  INPUT="1000" else
                        funkcija (9)      when  INPUT="1001" else
                        funkcija (10)     when  INPUT="1010" else
                        funkcija (11)     when  INPUT="1011" else
                        funkcija (12)     when  INPUT="1100" else
                        funkcija (13)     when  INPUT="1101" else
                        funkcija (14)     when  INPUT="1110" else
                        funkcija (15)     when  INPUT="1111";

      process(CLK, set, reset)
      begin
            if (CLK='1') and clk'event then
                  if  reset='1'then F<='0';
                        elsif set= '1' then F<='1';
                              elsif clock_enable='1' then F<=F_ASYNC;
                  end if;
            end if;
      end process;
end cell;
```

set i reset se koriste za uspostavljanje inicijalnog stanja, a CLK i clock_enable za generisanje stanja ćelije kontrolisano taktom. Kada se aktiviraju, reset i set blokiraju funkcionisanje ćelije. Signal reset ima prioritet u odnosu na set. Pri vrednosti '1', reset je aktiviran. Tada je na izlazu ćelije vrednost '0'. Pri aktivnom set, a isključenom reset ulazu izlaz ćelije dobija vrednost '1'. Ulazni signal koji takođe kontroliše izlazno stanje ćelije je clock_enable. Samo pri vrednosti '1' ovog ulaza (a kada su isključeni i reset i set) moguće je generisati izlazni signal na osnovu prenosne funkcije ulaznog podatka.

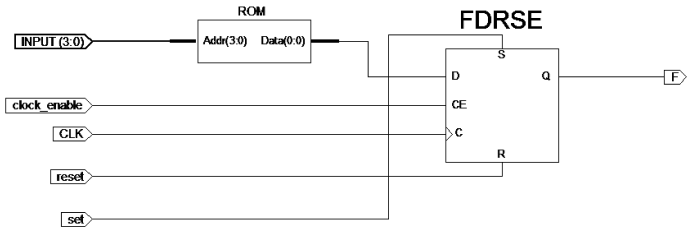
Velika pogodnost VHDL-a je što omogućava opis ponašanja modela (AbstractBehaviour Modeling), ali i opis strukture modela (Hardware Structure Modeling), što implicira samo ponašanje. Korišćen softverski alat omogućava računarsku simulaciju modela kreiranjem testbench koda. U testbench kodu se definišu ulazni signali sistema. Pokretanjem testbench-a računar simulira ponašanje sistema. Na taj način se mogu pratiti i analizirati promene vrednosti određenih izlaza u sistemu.

Nakon verifikacije VHDL koda, alat za sintezu taj kod pretvara u primitivne komponente korišćenog FPGA čipa. Potom sledi formiranje toka podataka sa opisom primitivnih komponenti i njihovog načina povezivanja, koji služi za konfigurisanje čipa po zadatom VHDL kodu.

Način na koji alat za sintezu izneti VHDL kod ćelije prevodi u primitivne komponente prikazan je na slici 2. Sa slike je uočljivo da se jedna ćelija implementira korišćenjem jednog LUT-a i jednog flip-flopa.

Princip rada generatora

Generator slučajnih brojeva u ovom slučaju radi na principu četvorosusednog automata sa asimetričnim konekcijama.

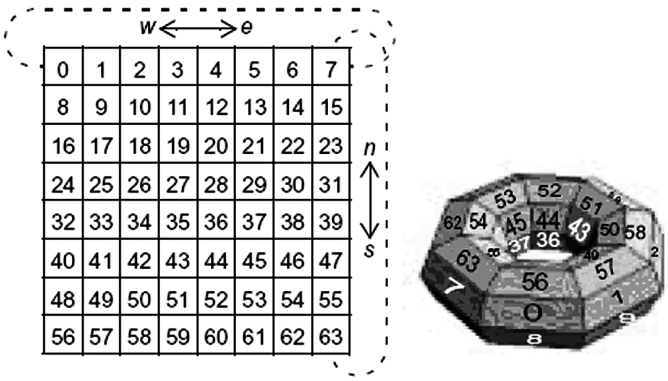


Slika 2. Realizacija jedne ćelije CA pomoću primitivnih komponenti čipa Spartan-3E

Četvorosusedni sistem podrazumeva da je svaka ćelija na određeni način povezana sa četiri ćelije koje uslovljavaju njeno stanje u sledećoj konfiguraciji. Susedstvo koje broji četiri ćelije omogućava veoma pogodnu implementaciju na hardveru zbog same arhitekture korišćenog čipa. Naime, programabilni blok (CLB, slika 1) koji je osnovna jedinica čipa sadrži generator funkcije (LUT), koji izvršava funkciju na osnovu četiri ulazna podatka. Na taj način je jedna ćelija automata implementirana tako da jedna jedinica programabilnog bloka sa svojim LUT generatorom vrši njenu funkciju. Kada bi ćelija imala pet suseda bila bi implementirana na dva LUT-a s tim da bi jedan od njih bio nepotpuno iskorišćen. To bi značajno povećalo broj zauzetih CLB-ova korišćenog čipa i smanjilo maksimalnu frekvenciju na kojoj čip stabilno radi.

Konekcija predstavlja način na koji su ćelije unutar automata povezane i definisana je na isti način za svaku ćeliju. Automat je predstavljen na slici 3. kao matrica numerisanih ćelija. Označeni pravci, na slici 3, W-E (West-East) i N-S (North-South) se koriste radi opisivanja konekcije. ćelije sa kojima je ostvarena konekcija su susedi date ćelije. Pozicije suseda se označavaju udaljenošću po pravcima W-E, N-S u odnosu na centralnu ćeliju. Na primer u realizovanom generatoru uspostavljena je konekcija (2N2W, c, N2E, 2SE), gde c predstavlja rekurzivnu vezu, tj. ćelija je sama sebi sused. Periferne ćelije koriste periodično ponavljanje vrsta i kolona matrice po pravcima (W-E, N-S) pri konekciji što povezanom automatu daje oblik torusa (slika 3). Na primer periferna ćelija 1 će biti povezana sa 55, 1, 59, 18.

Prenosna funkcija sistema se definiše na osnovu istinitonosne tablice. Primer je prikazan na slici 4. Ova tablica reguliše vrednost izlaza ćelije (tj. njenog stanja u narednoj konfiguraciji) u zavisnosti od ulaza (tj. vrednosti prethodnog stanja suseda). Oznaka generatora zavisi od ove funkcije. Očita se vrednost binarnog broja iz tablice, zatim se taj broj pretvori u dekadni koji predstavlja oznaku funkcije. Primer je predstavljen na slici 4 – oznaka automata je CA06990.



Slika 3.
Prikaz celularnog automata matricom numerisanih ćelija. Povezivanjem perifernih ćelija automat dobija oblik torusa.

Figure 3.

Kako svaka ćelija može imati dve vrednosti (0 ili1), ulazu u ćeliju može imati 2^4 različitih vrednosti. Za svaki ulaz prenosna funkcija definiše jednu od dve vrednosti (0 ili1) na izlazu iz ćelije. To znači da je moguć broj prenosnih funkcija u ovom slučaju 2^{16}. Kako sve funkcije ne daju zadovoljavajuće rezultate, potrebno je naći odgovarajuću.

Testiranje

Odgovarajuća prenosna funkcija tražena je eksperimentalno. Statističke osobine generatora ispitivane su DIEHARD baterijom testova. Rezultati DIEHARD testa su korišćeni kao finalni kriterijum ispravnosti generatora. Kako DIEHARD testiranje traje tridesetak sekundi na dostupnim računarima, a mogućih prenosnih funkcija ima 2^{16}, to bi ovakva provera trajala dvadesetak dana aktivnog testiranja. Iz tog razloga, bilo je potrebno eliminisati nepogodne kandidate za funkciju na neki drugi, brži način. To je urađeno određivanjem entropije uzorka.

Kako funkcija koje treba proveriti ima veoma mnogo, nije praktično svaku od njih implementirati na čip, pa zatim odatle učitavati uzorak brojeva. Iz tog razloga urađena je simulacija uređaja u programskom jeziku C++. Simulacija je koristila za generisanje uzorka brojeva za ispitivanje.

Nakon odabira zadovoljavajuće funkcije, VHDL kod je implementiran na Spartan-3E XC3500S-E. Korišćena je razvojna ploča Xilinx Spartan3-E Starter Kit (www.xlinx.com).

Čitav proces testiranja ponovljen je nad podacima dobijenim sa izlaza implementiranog kola.

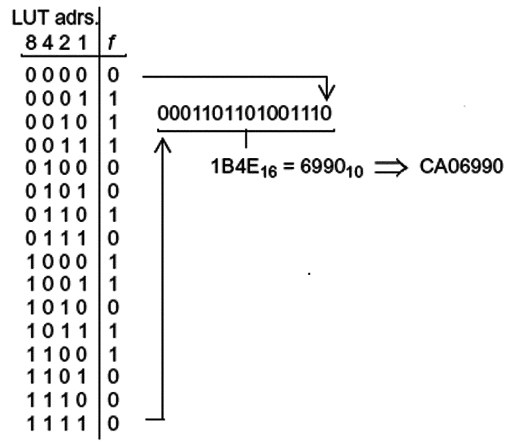
Entropija

Pojam entropije je zapravo matematički pokazatelj neuređenosti sistema (Belić 1990). Sam pojam slučajnosti podrazumeva visoku neuređenost, odnosno slučajna sekvenca brojeva bi trebalo da ima veoma visoku entropiju.

Formula za izračunavanje entropije sistema je sledeća:

$$S = -\sum_{i=0}^{2^{16}-1} p_i \log_2 p_i \qquad (1)$$

pri čemu je p_i verovatnoća ponavljanja broja i. Logaritam od nule je definisan kao nula u ovom slučaju. Entropiju na osnovu formule (1) je definisao naučnik Klod Šenon i to je tzv. informaciona ili Šenonova entropija.



Slika 4. Istinitosna tablica prenosne funkcije CA. Prenosna funkcija prevođenjem iz binarnog u dekadni sistem predstavlja oznaku automata

Sekvenca slučajnih brojeva bi trebalo da ima određenu raspodelu. U konkretnom slučaju to je uniformna raspodela. Kada se u formuli (1) p_i zamene teorijskim verovatnoćama za konkretnu raspodelu dobija se teorijska vrednost entropije. što je entropija sekvence bliža teorijskoj, veća je i verovatnoća da je raspodela odgovarajuća. Program koji računa entropiju napisan je u programskom jeziku C++.

DIEHARD

Ako su brojevi uniformno raspodeljeni, tj. jednako zastupljeni, ne mora značiti da ispunjavaju kriterijume slučajnosti. Oni takođe ne smeju biti u bilo kakvoj međusobnoj korelaciji, jer bi to u primeni dalo veoma loše rezultate.

Prof. Džordž Marsaglia (Florida State University), kreirao je DIEHARD bateriju testova (www.csis.hku.hk) koja sa velikom sigurnošću potvrđuje ili opovrgava uspešnost random sekvence. Sastoji se iz 18 nezavisnih statističkih testova. Ovi testovi se zasnivaju na nekim poznatim teoremema iz verovatnoće (paradoks rođendana, majmunska teorema, varijacije kockarskih igara...). Kada se sprovedu nad zadatim uzorkom, trebalo bi da se dobiju odgovarajuće verovatnoće ili raspodele (u zavis-

nosti od testa) ako je sekvenca zaista slučajna. Test vraća tzv. '*p*-vrednosti' koje su na intervalu (0,1). Ako je odstupanje rezultata od očekivane raspodele ili verovatnoće veliko, *p*-vrednosti su jako blizu 0 i 1 ($p > 0.9999$ ili $p < 0.0001$) i to znači da je generator pao taj test. Da bi generator prošao DIEHARD bateriju testova, mora proći i svaki pojedinačan test. Test zahteva uzorak od bar 3 000 000 32-bitnih brojeva. Kako su brojevi u konkretnom slučaju 64-bitni, jedna matrica generiše dva 32-bitna broja. Prvi čine neparni a drugi parni bitovi.

χ^2-test

Pirsonov χ^2 (hi-kvadrat) test (Mališić 1999) se koristi u matematičkoj statistici kao tzv. test saglasnosti. Testovi saglasnosti služe da bi se proverila određena hipoteza koja se odnosi na raspodelu uzorka. Hipoteza koja se proverava ovakvim testom je uglavnom neparametarska, odnosno odnosi se samo na pretpostavku tipa raspodele. Generisani slučajni brojevi trebalo bi da budu jednako zastupljeni, tj. da raspodela generisanog uzorka bude uniformna. Testom se proverava u kolikoj meri se raspodela dobijene sekvence slaže sa pretpostavljenom:

$$P(F(X) = F_o(X)) = 1 - \alpha,$$

pri čemu je $1 - \alpha$ verovatnoća da je hipoteza tačna pri određenim vrednostima veličine χ^2.

Vrednost statistike χ^2 se uzima u obzir kao mera odstupanja empirijske od hipotetičke raspodele:

$$\chi^2 = \sum_{j=1}^{r} \frac{m_j - f_j}{f_j}$$

pri čemu je f_t frekvencija svakog broja i iz uzorka po hipotetičkoj raspodeli, m_i takođe frekvencija broja i, ali eksperimentalno određena. Očigledno je da je χ^2 neka vrsta mere odstupanja eksperimentalne frekvencije od teorijske, pa manje vrednosti ove veličine idu u prilog hipotezi. Granica veličine χ^2 koja ne protivreči hipotezi se određuje na sledeći način. Iz tablice za χ^2 raspodelu se očita broj ε za koji važi:

$$P\{\chi^2_{r-l-1} \mid \varepsilon\} = \alpha$$

i koji zavisi od veličina $r - l - 1$ i α. (l je broj nepoznatih parametara pretpostavljene raspodele, pri uniformnoj raspodeli $l = 2$; r je broj različitih vrednosti u uzorku). To znači da je α verovatnoća pripadanja veličine χ^2 tzv. kritičnoj oblasti $[\varepsilon, +\infty)$. Veličina α (prag značajnosti) se uzima tako da ima što manju vrednost (konkretno $\alpha = 0.05$), što znači da u slučaju da eksperimentalna vrednost χ^2 pripada kritičnoj oblasti, hipoteza se odbacuje jer je verovatnoća za to jako mala (0.05). U suprotnom, hipoteza se prihvata.

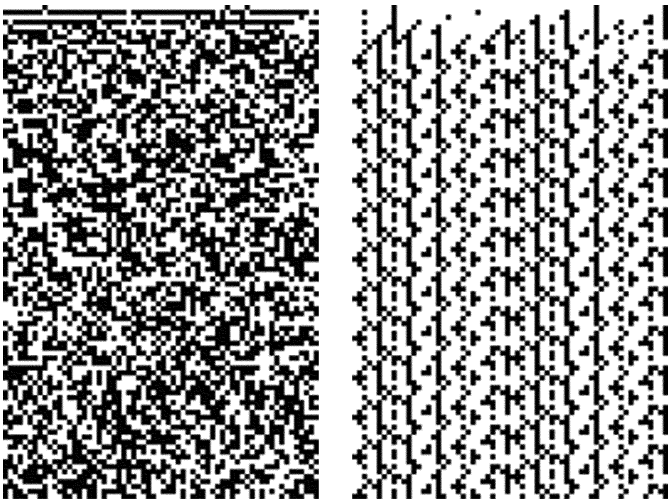
Rezultati testiranja

Uzorak za izračunavanje entropije čine šesnaestobitni brojevi. Prvi šesnaestobitni broj je definisan od prve do šesnaeste ćelije. Sledeći se dobija bitskim pomeranjem unutar matrice za jednu ćeliju. Dakle sledeći broj uzorka je definisan od 2. do 17. ćelije, a poslednji od 64. do 15. To je ukupno 64 broja u uzorku. Koristi se takođe i prethodnih 16 generacija matrice da bi se izdvojila još 64 broja uzorka. Naime, svaka pojedinačna ćelija formira po jedan broj tako što njenih 16 prethodnih stanja čini po jedan bit tog broja. Zatim se generiše nova matrica koja na isti način daje 128 (64 + 64) brojeva uzorka.

Pre početnog stanja CA koje se koristi u izračunavanju entropije, generiše se 80 iteracija (64 za dostizanje stanja visoke entropije i 16 koji se koriste za inicijalno izračunavanje). Uzorak je kompletan nakon još 2^{13} ciklusa. Na taj način, dobijen je uzorak koji čini ukupno $2^{13} \cdot 128 = 2^{20}$ brojeva. Ako je sekvenca slučajna, trebalo bi da svi brojevi budu jednako zastupljeni. Kako postoji maksimalno 2^{16} šesnaestobitnih brojeva, idealno bi bilo da je svaki zastupljen $2^{20} / 2^{16} = 2^4$ puta. To bi značilo da je idealna verovatnoća ponavljanja svakog broja $1/2^4$. Ako se u formuli (1) zamene idealne vrednosti verovatnoće za svaki broj ($1/2^4$), dobija se da je vrednost entropije u idealnom slučaju 16.

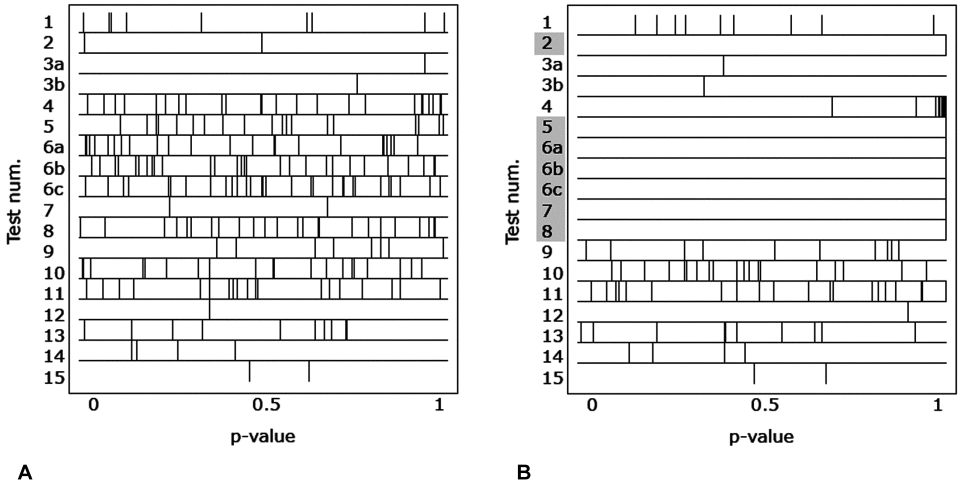
Na kraju je izabrano 200 kandidata sa najvišom entropijom koji su dalje podvrgnuti DIEHARD testovima. Od prvih 200 funkcija najviše entropije samo je nekoliko prošlo DIEHARD bateriju testova: CA18, CA24, CA171, CA27225... Nisu svi uzorci sa najvišim entropijama prošli test, ali kako entropija opada koncentracija odgovarajućih generatora je sve manja.

Na slici 5A prikazana je grafička interpretacija sekvence sa visokom, a na slici 5B sa niskom en-



Slika 5.
Grafička interpretacija sekvence brojeva generisane automatom CA27225 visoke entropije (A) automatom CA49278 niske entropije (B)

Figure 5.



Slika 6. Grafička interpretacija sekvence brojeva generisane automatom CA49278 niske entropije (A) i automatom CA38039 koja ne prolazi testiranje (B)

Figure 6.

tropijom. Prva sekvenca je generisana automatom CA27225, a druga automatom CA49278. U svakoj vrsti je predstavljen po jedan 64-bitni generisan broj u binarnom brojevnom sistemu, tako da crna tačka predstavlja 1, a bela 0. Na taj način je prikazano prvih 200 brojeva.

Rezultati DIEHARD baterije testova takođe se mogu prikazati grafički – ucrtavanjem p vrednosti na brojnu osu. Svaki od 18 testova vraća više p vrednosti. Grafički prikaz rezultata za sekvencu koja prolazi DIEHARD testiranje je dat na slici 6A, a za sekvencu koja ne prolazi na slici 6B. Sivom bojom označeni su testovi na kojima generator slučajnih brojeva nije prošao.

Nad listama brojeva generatora koji su prošli do sada pomenuta testiranja, urađen je i χ^2 test, kao jedan od standardnih testova saglasnosti. U konkretnom testiranju, uzet je prag značajnosti $\alpha = 0.05$, dok je tablična vrednost $\varepsilon = 33205$. Rezultati svih testiranja generatora imaju dobre rezultate, odnosno vrednosti χ^2 koje su manje od ε. Dobijene vrednosti χ^2 u nekim slučajevima su:

CA18: $\chi^2 = 32332$

CA171: $\chi^2 = 32768$

CA28: $\chi^2 = 33205$

Kombinovan CA (18, 24, 28,171): $\chi^2 = 32619$.

Diskusija

Ispravnost simulacije VHDL koda generatora je potvrđena kreiranim testbench fajlom. Nakon toga uređaj je implementiran na FPGA kolo Xilinx FPGA Spartan 3E. Rezulati C simulacije i izlaz sa čipa se poklapaju.

Osim izbora funkcije i način povezivanja utiče na vrednovanje generatora, tj. ne može bilo koja konekcija biti ključ za generisanje kvalitetne sekvence pseudoslučajnih brojeva. Eksperimentalno je utvrđeno da nijedna od jednostavnih simetričnih konekcija koje su testirane ne prolazi DIEHARD test (npr. (N, c, W, E)). Rekurzivna veza (c) značajno doprinosi neuređenosti sistema.

Maksimalna frekvencija generatora na čipu je oko 250MHz, što je, uzimajući u obzir karakteristike Spartan-3E čipa, veoma visoka vrednost. Glavni razlog za ovakav rezultat je jednostavna implementacija same ćelije CA, jer kritična putanja signala u sistemu prolazi kroz samo jedan LUT element. Međutim brzina ipak zavisi od dinamičkih osobina sistema. Realizacija kompleksnih konekcija prouzrokuje smanjenje maksimalne frekvencije sistema. Da bi se proverilo koliko ovaj faktor utiče na brzinu sistema potrebno je implementirati i testirati nekoliko CA sa različitim konekcijama.

Pri generisanju brojeva, ispitan je i 'kombinovan generator'. Uzete su random liste četiri generatora koja prolaze DIEHARD test (CA18, CA24, CA28, CA171). Formirana je nova lista u koju se periodično upisuje po jedan broj iz svake od ove četiri liste. Ispostavilo se da i ovako generisana sekvenca brojeva prolazi DIEHARD. Implementiranjem sva četiri generatora na čip, pri jednom taktu bi se dobila četiri broja sekvence. To naravno povećava brzinu generisanja četiri puta.

U tabeli 1 prikazano je nekoliko parametara po kojima se generator može uporediti sa random bibliotekama C++ i MATLAB, testiranim na računaru Athlon64 3000+. Generator ima znatno veću prednost u pogledu brzine generisanja, a za nijansu je bolji i u pogledu vrednosti χ^2 testa. C++ biblioteka nije mogla biti testirana DIEHARD testom jer generiše 15-bitne brojeve, dok DIEHARD zahteva 32-bitne.

Tabela 1. Prikaz karakteristika pseudoslučajnih generatora

	DIEHARD	Vreme generisanja jednog broja (ms)	χ^2 ($\varepsilon = 33205$)
C++	–	10.7	32756
Matlab	+	45.6	32664
CA 18	+	3.989	32332
Kombinovan CA	+	0.997	32619

Što se tiče perioda generatora, u dosadašnjim ispitivanjima nije određen. Teorijski bi maksimalno mogao biti 2^{64}. Kada bi se ponovila inicijalna vrednost, cela sekvenca bi bila ponovljena. Činjenica da izdvojeni generatori imaju entropiju blisku teorijskoj vrednosti, kao i da prolaze DIEHARD test, implicira da im je i period veoma veliki. Eksperimentalno nije dokazano, ali postoji teorijska opasnost skraćivanja perioda, odnosno upadanja u petlju. Moguće je da više konfiguracija CA generiše istu matricu. Dakle, postoji mogućnost da se neki od brojeva iz sekvence ponovi pre inicijalne vrednosti. To bi značilo da se sekvenca periodično ponavlja od prvog ponovljenog broja u nizu, čime je period skraćen.

Zaključak

Generator slučajnih brojeva predstavljen u ovom radu poseduje dobre karakteristike u pogledu brzine generisanja sekvence, a odlikuje se i malom potrošjom. Prolaznost DIEHARD baterije testova potvrđuje dobre statističke osobine generisanih brojeva. U daljem radu moguće je istražiti osobine generatora drugačijih osobina po pitanju dimenzije,

konekcije, tačne brzine izvršavanja na hardveru, perioda.

Zahvalnost. Zahvalila bih se prvenstveno Đorđu Nijemčeviću i Miodragu Đukiću na idejama i pomoći pri realizaciji rada. Zatim saradnicima i polaznicima seminara Elektronike na savetima i pruženoj podršci.

Literatura

Belić M. R. 1990. *Deterministički haos*. Beograd: Institut za Fiziku, Sveske fizičkih nauka

Mališić J. 1999. *Verovatnoća i matematička statistika*. Beograd: Krug.

Perry D. L. 2002. *VHDL Programming by Example*. McGraw-Hill.

Shackleford B., Tanaka M., Carter R. J., Snider G. 2001. FPGA Implementation of Neighborhood-of-Four Cellular Automata Random Number Generators. www. hpl. hp. com/techreports/2001/HPL-2001-290.pdf

Wolfram S. 1983. *Statistical mechanics of cellular automata*. Stephenwolfram.com.

www.csis.hku.hk/čdiehard/cdrom/

www.xilinx.com/xlnx/xebiz/designResources/ip_product_details.jsp?key=HW-SPAR3E-SK-US

Dajana Danilović

APSTRAKT NA ENGL