

## CMOS IMPLEMENTACIJA HOPFILDOVE NEURONSKE MREŽE

Miljan S. Nikolić, Vančo B. Litovski  
Univerzitet u Nišu, Elektronski fakultet, Beogradska 14, 18000 Niš

**Sadržaj** – Veštačke neuronske mreže uspešno se koriste za rešavanje problema kod kojih standardne metode ne daju zadovoljavajuće rezultate. Sastoje se od više međusobno povezanih veštačkih neurona. U ovom radu opisane su dve implementacije Hopfildove neuronske mreže. Prva je realizovana na Xilinx FPGA čipu. Drugom implementacijom došlo se do nivoa lejauta korišćenjem programskog paketa Tanner

### 1. UVOD

Kao model za realizaciju veštačkih neuronskih mreža koristi se naravno nervni sistem živih bića, pre svega čoveka. Nervni sistem se pokazao superiornim kod rešavanja problema, kao što su prepoznavanja govora pri različitim intenzitetu govornika, ili prepoznavanje likova bez obzira na orijentaciju, osvetljenje, i u poređenju sa savremenim računarima. Efikasnost nervnog sistema ogleda se u masovnom paralelizmu njegovih osnovnih procesnih jedinica tj. neurona.

### 2. OSOBINE HOPFILDOVE NEURONSKE MREŽE

Hopfildova neuronska mreža pripada tipu rekurentnih mreža. Ova mreža se može iskoristiti za realizaciju nelinearne asocijativne memorije tj. memorije adresirane sadržajem. Pored mogućnosti adresiranja sadržajem Hopfildova neuronska mreža ima svojstvo ispravljanja grešaka, i pored prisustava pogrešnih bitova na ulazu mreže na izlazu možemo dobiti tačnu informaciju. Hopfildova neuronska mreža nema fazu obučavanja, već se sinaptički koeficijenti mogu odrediti računskim putem.

Pored željenih osobina Hopfildova neuronska mreža ima i nedostatke, prvi je da mreža nije apsolutno stabilna. Pored potencijalne nestabilnosti Hopfildova neuronska mreža ima svojstvo da je komplement stabilnog stanja takođe stabilno stanje, a takođe i linearna kombinacija neparnog broja željenih stabilnih stanja je stabilno stanje, ali se ovaj fenomen javlja kod mreža sa velikim brojem neurona. To je posledica simetrije mreže. Ova dva fenomena degradiraju efikasnost Hopfildove mreže za realizaciju memorije adresirane sadržajem.

Mera stabilnosti mreže je odnos signal-šum. Na osnovu [1] odnos signal-šum je

$$\rho = N/(p-1) \quad (1)$$

gde je  $p$  broj stabilnih stanja, a  $N$  broj neurona. Mreža će biti stabilna samo ako je odnos signal-šum veliki. Sve dok mreža nije preopterećena, tj. sve dok je broj stabilnih stanja  $p$  mali u poređenju sa brojem neurona  $N$  mreža je stabilna.

Važan parametar mreže je njen kapacitet, odnosno koliki broj stabilnih stanja mreža može da ima, a da pri tome odziv mreže ima zadovoljavajuću tačnost. Da bi

mreža bila u stanju da da ispravan odziv sa odgovarajućom tačnošću na osnovu [1] potrebno je da je

$$\rho_{min} = 2 \ln(N), \quad (2)$$

tako da je maksimalni kapacitet mreže

$$p_{max} = N/\rho_{min} \text{ tj. } p_{max} = N/2 \ln(N). \quad (3)$$

Ukoliko želimo da je verovatnoća ispravnog odziva 99% dobija se strožiji uslov za maksimalni kapacitet mreže  $p_{max}$  tj.

$$p_{max} = N/4 \ln(N). \quad (4)$$

Bez obzira koji od dva navedena uslova za određivanje maksimalnog kapaciteta koristimo, moramo održavati broj stabilnih stanja dovoljno malim da važe navedene relacije za određivanje  $p_{max}$  kako bismo mogli dobiti ispravnu reč. Ovo značajno ograničava efikasnost Hopfildove mreže.

### 3. PROJEKTOVANJE HOPFILDOVE MREŽE

Za realizaciju Hopfildove neuronske mreže osnovna procesna jedinica je neuron. On ima dva stanja zavisno od aktivacionog potencijala koji deluje na njega. U "aktivnom" stanju izlazna vrednost neurona  $i$  je  $s_i = +1$ , dok je "isključenom" stanju izlazna vrednost neurona  $i$   $s_i = -1$ . Za mrežu od  $N$  neurona, stanje mreže je određeno vektorom  $s = [s_1, s_2, \dots, s_n]^T$ . Kako stanje neurona ima dve vrednosti  $s = \pm 1$  potreban je jedan informacioni bit za njegovo pamćenje, a za neuronsku mrežu sa  $N$  neurona potrebna je informaciona reč od  $N$  bitova koja su predstavljena vektorom dimenzija  $N \times 1$ .

Veza neurona  $i$  i  $j$  modeluje se sinaptičkom težinom  $w_{ij}$ , koja predstavlja uticaj izlaza neurona  $i$  na aktivacioni potencijal neurona  $j$ . Uticaj može biti pozitivan (ekscitaciona sinapsa) ili negativan (inhibitorna sinapsa). Potencijal  $v_j$  koji deluje na neuron  $j$  je superpozicija svih potencijala koji deluju na njega. Važi sledeća relacija:

$$v_j = \sum_{i=1}^N w_{ji} s_i - \theta_j; \quad (5)$$

gde je  $\theta_j$  prag neurona  $j$ . Prema tome, neuron  $j$  menja stanje po sledećem zakonu:

$$s_j = \begin{cases} +1, & \text{za } v_j > 0 \\ -1, & \text{za } v_j < 0 \end{cases}, \quad (6)$$

ili u kompaktnijoj formi

$$s_j = \text{sgn}(v_j). \quad (7)$$

Ostaje samo pitanje stanja neurona ako je potencijal  $v_j$  jednak nuli. U tom slučaju stanje neurona se ne menja, tj. ostaje isto kao u prethodnom trenutku.

Postoje dve faze u radu Hopfildove mreže: faza pamćenja i faza pribavljanja informacije.

**Faza pamćenja.** Pretpostavimo da želimo da zapamtimo skup od  $p$   $N$ -dimenzionalnih vektora (binarnih reči) datih sa:  $\{ \xi_\mu / \mu = 1, 2, \dots, p \}$ . Ovih  $p$  vektora predstavljaju

memoriju, koja će biti “sadržana” u mreži. Neka  $\xi_{\mu,i}$  označava  $i$ -ti element stabilnog stanja  $\xi_{\mu}$ . Prema generalizaciji Hebovog postulata učenja [1] sinaptičke težine su definisane sa:

$$w_{ji} = \frac{1}{N} \sum_{\mu=1}^p \xi_{\mu,j} \xi_{\mu,i} \quad (8)$$

Svrha konstante  $1/N$  je pojednostavljenje matematičkog opisa pribavljanja informacija. Pri tome je  $w_{ii}=0$  za svako  $i$ , što znači da neuroni namaju samo-reakciju. Ako sa  $\mathbf{W}$  označimo matricu sinaptičkih težina dimenzija  $\mathbf{N} \times \mathbf{N}$ , iskoristimo relacije za  $w_{ji}$  i da su elementi glavne dijagonale jednaki nuli, možemo da napišemo sledeću relaciju za matricu sinaptičkih težina:

$$\mathbf{W} = \frac{1}{N} \sum_{\mu=1}^p \xi_{\mu} \xi_{\mu}^T - \frac{\mathbf{P}}{N} \quad (9)$$

Matrica sinaptičkih težina je simetrična matrica tj.

$\mathbf{W} = \mathbf{W}^T$ , izlaz svakog neurona se vodi na ulaz svih ostalih osim na sopstveni.

*Faza pribavljanja.* Za vreme ove faze dovodimo  $N$ -dimenzionalni vektor  $\mathbf{x}$  na ulaz mreže koji predstavlja početno stanje mreže. Mreža menja stanje dinamički prema pravilu da svaki neuron slučajno ali sa određenom frekvencijom ispituje svoj aktivacioni potencijal i saglasno njegovoj vrednosti ažurira svoje stanje. Ažuriranje stanja iz iteracije u iteraciju je determinističko, ali je selekcija neurona koji vrši ažuriranje slučajna. Ažuriranje se vrši sve dok mreža ne dođe u stabilno stanje tj. vremenski invarijantno stanje, odnosno ako se zadovolji relacija

$$\mathbf{x} = \text{sgn}(\mathbf{W}\mathbf{x} - \boldsymbol{\theta}) \quad (10)$$

Ovakav način ažuriranja se zove *asinhron*.

*Sinhroni* model (Little i Show) koristi istu matricu sinaptičkih težina kao i Hopfildov model, razlika je u načinu promene stanja. Kao što i sam naziv govori stanja se ažuriraju sinhrono. Razlika u načinu ažuriranja ima uticaj na broj iteracija potrebnih da mreža ode u stabilno stanje.

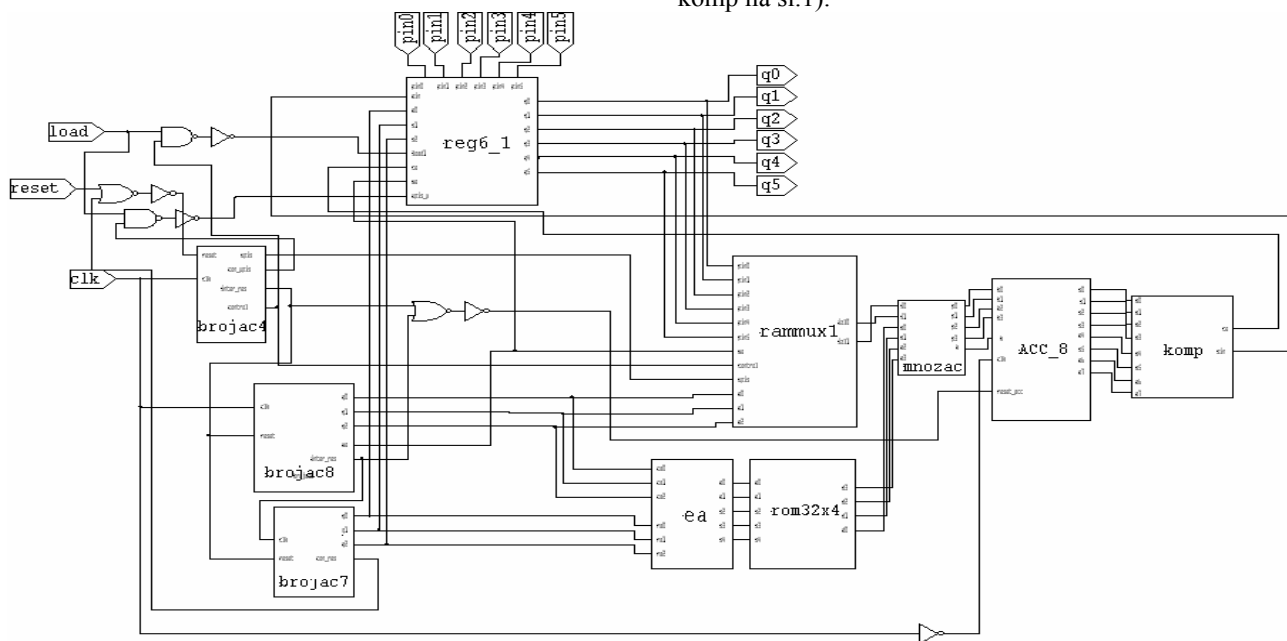
#### 4. IZRADA PROTOTIPA HOPFILDOVE NEURONSKE MREŽE

Realizovana je Hopfildova mreža koju čini šest neurona. Izabrana stabilna stanja su :  $\xi_1=(-1,-1,-1,-1, 1, 1)$ ,  $\xi_2=(-1,-1, 1,-1, 1, 1)$ ,  $\xi_3=(-1, 1,-1,-1, 1, 1)$ ,  $\xi_4=(-1, 1, 1,-1, 1, 1)$ . Na osnovu osobine da je komplement stabilnog stanja takođe stabilno stanje dobijamo sledeće vektore stabilnih stanja:  $\xi_5=( 1, 1, 1, 1,-1,-1)$ ,  $\xi_6=( 1, 1,-1, 1,-1,-1)$ ,  $\xi_7=( 1,-1, 1, 1,-1,-1)$ ,  $\xi_8=( 1,-1,-1, 1,-1,-1)$ . Na osnovu relacije (9) matrica sinaptičkih koeficijenata je data izrazom (11). Koeficijent  $1/N$  izostavljen je jer se koristi celobrojna aritmetika.

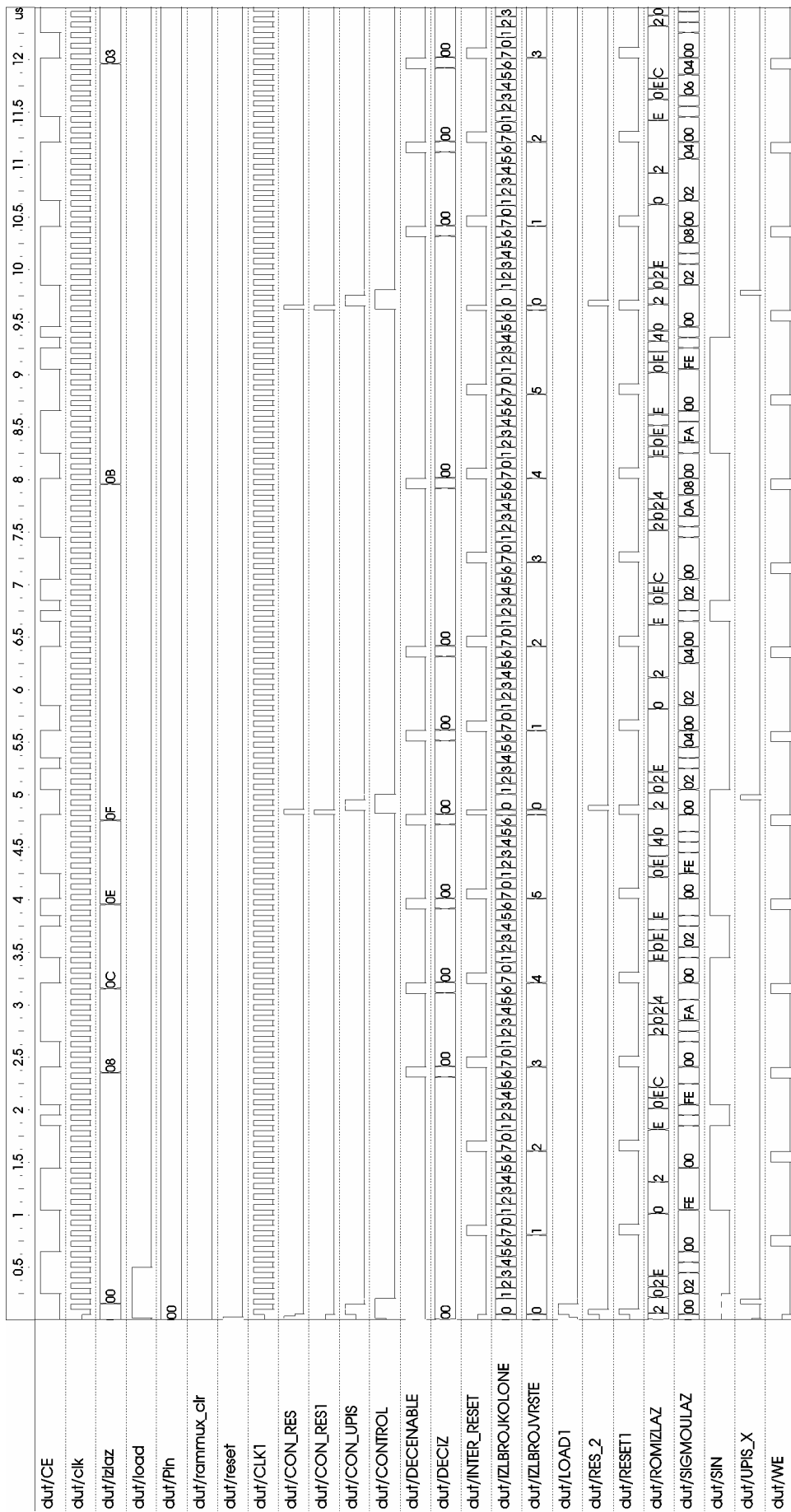
Kolo je opisano i simulirano ACTIVE-HDL programskim paketom i nakon verifikacije VHDL opis kola je importovan u Xilinx Foundation Project Manager kojim je izvršeno programiranje FPGA čipa XC4003.

Korišćenjem strukturnog VHDL opisa bilo je moguće implementirati kolo korišćenjem Tanner programskog paketa.

Šema realizovanog kola je prikazana na slici 1. Kolo je sinhrono i sastoji se od ulazno/izlaznog stacionarnog šestobitnog registra (reg6\_1 sl.1), kontrolne logike (brojac4 sl.1), ROM-a 32x4 u kome su upisani sinaptički koeficijenti (ROM 32x4 sl.1), pomeračkog šestobitnog registra u kome se čuva stanje izlaza iz prethodne iteracije i dodatne logike koja upravlja tokom izračunavanja (blok rammux1 sl.1), brojača osnova brojanja sedam i osam (brojac7 i brojac8 na sl.1 respektivno) koji adresiraju vrstu i kolonu matrice sinaptičkih koeficijenata tj. ROM-a, adresa se izračunava u bloku ea (sl.1), množača akumulatora i komparatora (blokovi mnozac, acc\_8 i komp na sl.1).



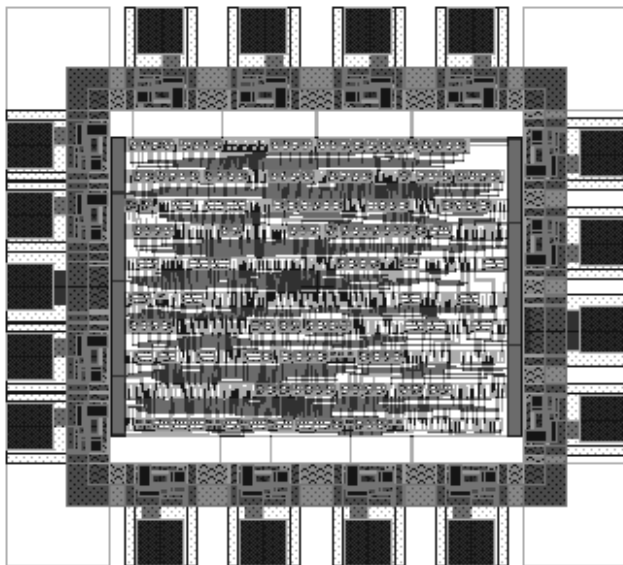
sl.1 Blok šema realizovanog kola



sl.3 Rezultati simulacije

$$W = \begin{bmatrix} 0 & 2 & 0 & 2 & -2 & -2 \\ 2 & 0 & -2 & 0 & 0 & 0 \\ 0 & -2 & 0 & 2 & -2 & -2 \\ 2 & 0 & 2 & 0 & -4 & -4 \\ -2 & 0 & -2 & -4 & 0 & 4 \\ -2 & 0 & -2 & -4 & 4 & 0 \end{bmatrix}. \quad (11)$$

Digitalna pobudna reč dovodi se na ulaz šestobitnog registra (reg6\_1 sl.1) i ukoliko je aktivan signal upisa (load sl.1), a istovremeno neaktivan signal reset sinhrono sa impulsom kontrolne logike (brojac4 sl.1) i taktim signalom izvršice se upis u reg6\_1. Sledećim taktim impulsom ulazna reč se upisuje u pomerački registar rammux1 (sl.1). Kontrolna logika takođe generiše reset signal koji resetuje sadržaj akumulatora (svi pragovi neurona su jednaki nuli), brojac7 i brojac8 na nulu. Brojac8 adresira kolonu matrice sinaptičkih koeficijenata, dok brojac7 adresira vrstu matrice sinaptičkih koeficijenata. Na osnovu vrednosti brojača (brojac7 i brojac8) u bloku ea se izračunava adresa ROM-a kojoj se pristupa. Izlaz ROM-a se vodi na ulaz množača gde se množi sa odgovarajućim stanjem neurona iz prethodne iteracije. Međurezultat se pamti u akumulatoru i kada se završi izračunavanje za određeni neuron izlaz komparatora, što predstavlja novo stanje neurona se upisuje u odgovarajući DFF reg6\_1. Nakon toga se resetuje akumulator, brojac7, brojac8 i izvrši šiftovanje sadržaja registra rammux1. Ova sekvenca se ponavlja sve dok kolo ne ode u stabilno stanje.



Sl.2 Lejaut realizovanog kola

Rezultati dobijeni simulacijom VHDL opisa kola Active HDL programskim paketom prikazani su na slici 3. Ulazna digitalna reč je  $x_0=[000000]$  (stanju neurona -1 odgovara logička nula, a stanju +1 odgovara logička jedinica) signal dut/Pin na slici 3. Nakon prve iteracije stanje mreže je  $x_1=[001111]$  ili u heksadecimalnoj notaciji 0F (signal dut/Izlaz na slici 3), nakon druge iteracije stanje mreže je  $x_2=[001011]$  ili 0B. Po završetku treće iteracije mreža prelazi u stabilno stanje  $x_3=[000011]$  ili 03 u heksadecimalnoj notaciji.

## 5. ZAKLJUČAK

Kolo je implementirano na XILINX FPGA čipu XC4003. Kolo zauzima 61 ćeliju od 100. Za jednu iteraciju

potrebno je 56 taktova, i kolo je testirano sa taktom od 8 MHz (najviši dostupan na čipu XC4003). Potrebno je maksimalno tri iteracije da bi se dostiglo stabilno stanje. Prilikom realizacije je uzeto da su svi pragovi neurona jednaki nuli, mada realizacija da pragovi budu proizvoljni zahteva dodatni ROM za pamćenje istih i da se signal za reset akumulatora iskoristi za postavljanje početne vrednosti akumulatora, koja treba da bude jednaka pragu neurona čije se stanje ažurira.

Ukoliko bi se povećao broj neurona pored povećanja hardvera (potreban je veći ROM za pamćenje matrice sinaptičkih koeficijenata) dolazi do smanjenja brzine rada kola usled povećanja broja taktova za realizaciju jedne iteracije.

Realizacija kola upotrebom programskog paketa Tanner urađena je u standardnoj CMOS Mosis 0.5 mikronske tehnologiji. Lejaut kola je prikazan na slici 2. Jezgro čipa je dimenzija  $883.75 \times 645.75 \mu\text{m}^2$ , dok ceo čip zauzima  $1333.15 \times 1204.7 \mu\text{m}^2$ . Kolo je uspešno prošlo DRC i LVS proveru.

## LITERATURA

- [1] S.Haykin, "Neural networks", *Macmillan College Publishing Company*, 1994.
- [2] S.Lj.Milenković, "Veštačke neuronske mreže", *Zadužbina Andrejević*, biblioteka *Dissertatio*, Beograd 1997.
- [3] B.W.Wah, "Special Issue on Artificial Neural Networks - Guest Editor's Introduction", *IEEE Trans. on Computers*, Vol. 40, No. 12, pp. 1317-1319, December 1991.
- [4] A.F.Murray, "Silicon implementations of neural networks", *IEE Proc.-F*, Vol. 138, No. 1, pp. 3-12, February 1991.
- [5] V.B.Litovski, "Projektovanje elektronskih kola: simulacija, optimizacija, testiranje, fizičko projektovanje", *DGIP "Nova Jugoslavija"*, Vranje, 2000.
- [6] V.B.Litovski, J.Radjenović, Ž.Mrčarica, S.Milenković, "MOS transistor modelling using neural network", *Electronics Letters*, Lett. 28(13), pp. 1766-1768, 1992.
- [7] V.B.Litovski, Ž.Mrčarica, T.Ilić, "Simulation of non-linear magnetic circuits modelled using artificial neural network", *Simulation Practice and Theory*, No. 5, pp. 553-570, 1997.
- [8] Active-HDL, ver. 3.6., ALDEC Inc., 1998.
- [9] <http://www.xilinx.com>
- [10] Tanner, ver. 8.0, Tanner EDA, 1999.

**Abstract:** Artificial neural networks are successfully used for solving the problems where standard methods cannot be applied. They consist of a number of mutually highly connected neurons. In this paper are presented two implementations of Hopfield's neural network.

First implementation is done using Xilinx FPGA chip. Second implementation is done to the layout level by Tanner software.

## IMPLEMENTATION OF HOPFIELD'S NEURON NETWORK

Miljan S. Nikolić and Vančo B. Litovski