Zahvala

Zahvaljujem mentoru prof. dr. sc. Adrijanu Bariću na podršci kroz projekt. Zahvaljujem što mi je uvijek bio dostupan i na snažnom poticaju za daljnjim radom i napretkom. Također bih volio zahvaliti dr. sc. Jurici Kundrati koji mi je pomogao preći preko svakog problema na kojeg sam naišao, ali me također tjerao da sve bitne dijelove prođem sam.

Sadržaj

Uvod

Razvoj strojnog učenja i, od nedavno, dubokog učenja doveo je do porasta korištenja neuronskih mreža. Ta područja umjetne inteligencije postaju nezaobilazna zbog velikog broja istraživanja u područjima kao što su raspoznavanje uzoraka, obrada jezika i robotika. Mnogo složenije arhitekture strojnog učenja, rezultat su sve veće računalne mogućnosti današnjih računala i ogromne količine raspoloživih podataka. Također, algoritmi dubokog učenja dobivaju na popularnosti kod uređaja poput senzora ili aktuatora koji su neophodni u mnogim zadacima poput klasifikacije slike ili prepoznavanja govora. Kao posljedica toga, sve je veći interes za primjenu neuronskih mreža na procesorima i mikrokontrolerima [1].

U ovom radu ćemo izvesti napad na neuronsku mrežu i pokušati izvući parametre mreže. Za primjer mreže ćemo koristiti jednoslojnu umjetnu neuronsku mrežu implementiranu na FPGA sklopu.

Mnogo je razloga da zaštitimo arhitekturu i parametre neuronske mreže. Jedan od razloga je to što se modeli strojnog učenja koji se koriste za vođenje medicinskih tretmana često temelje na pacijentovom genotipu, što čini ovo izuzetno osjetljivim iz perspektive privatnosti. Isto tako pribavljanje korisnih informacija o arhitekturi neuronske mreže može pomoći u saznavanju poslovnih tajni od konkurencije, što bi moglo dovesti do korištenja konkurentnih proizvoda bez kršenja prava intelektualnog vlasništva [1]. S obzirom na navedene razloge potrebno je pronaći odgovarajući način zaštite, a prvi korak je upoznavanje sa samim rizicima.

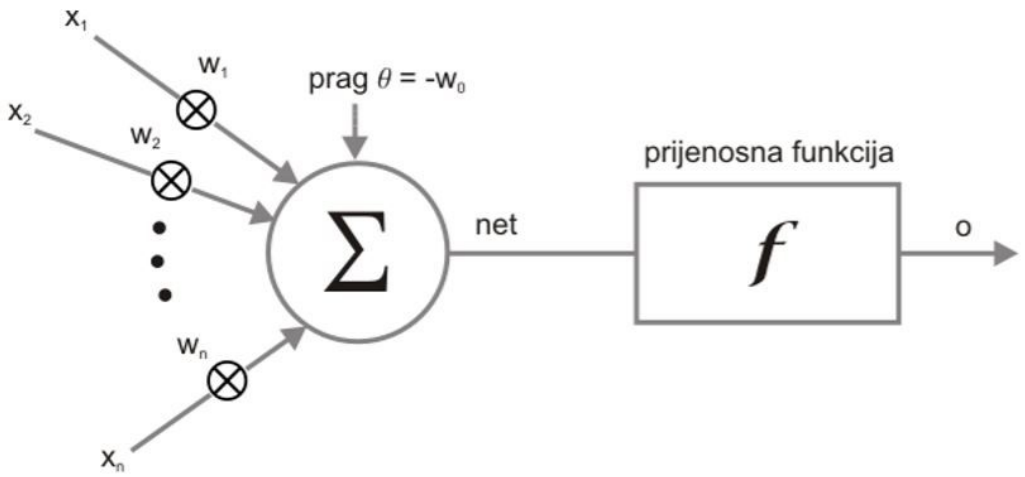
Digitalni sustavi mogu biti sigurnosno ranjivi na razini programske i sklopovske podrške. Napadi fokusirani na ranjivosti sklopovske podrške su tzv. *side – channel* napadi. Oni napadaju računalni sustav na temelju informacija kao što su vrijeme trajanja operacije, potrošnja energije, elektromagnetno zračenje pa čak i zvuk [2].

Tajne informacije iz računalnih sustava mogu procuriti preko mjerenja napajanja prilikom izvršavanja operacija, a u slučaju digitalnih sklopova one ovise o broju bitova koji mijenjaju svoje stanje kroz određeno vrijeme. Promjene stanja izazivaju punjenje i pražnjenje parazitskih kapaciteta koji crpe struju iz napajanja te stvaraju smetnje. Te smetnje mogu dati uvid u operacije digitalnog sklopa [3].

Cilj ovog rada je izvući parametre neuronske mreže procesom reverznog inženjerstva koristeći napad korelacijskom analizom napajanja (engl. *correlation power analysis*) . Reverzno inženjerstvo je jedan od problema neuronskih mreža implementiranih na digitalnom uređaju [1]. Porast korištenja neuronskih mreža na širokim područjima te činjenica da često sadrže tajne informacije je jedan od glavnih motiva da se bolje upoznamo s mogućim propustima sklopovske podrške.

1. Umjetne neuronske mreže

„Umjetna neuronska mreža je skup međusobno povezanih jednostavnih procesnih elemenata (neurona) čija se funkcionalnost temelji na biološkom neuronu i koji služe distribuiranoj paralelnoj obradi podataka“ po definiciji iz [4]. Postoje dvije faze rada s umjetnim neuronskim mrežama; faza učenja i faza obrade podataka. Učenje je iterativan postupak predočavanja ulaznih primjera i eventualno očekivanog izlaza pri čemu dolazi do postupnog prilagođavanja težina veza između neurona [5]. Na slici (Sl. 1.1) preuzetoj s [4] vidi se poseban slučaj umjetne neuronske mreže, perceptron, odnosno jedan umjetni neuron koji predstavlja jednoslojnu neuronsku mrežu. Vrijednosti s ulaza množi s odgovarajućim težinama i akumulira u tijelu. Toj se sumi dodaje pomak *w0,* te se zajedno propušta kroz prijenosnu funkciju na izlaz. U nastavku rada promatra se isključivo taj primjer umjetne neuronske mreže.



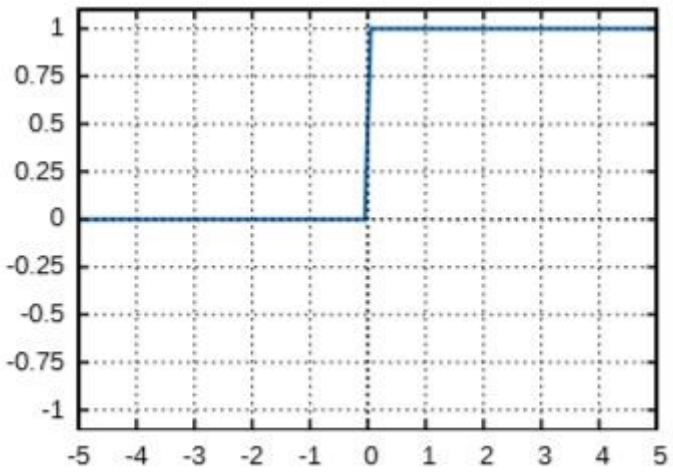
Sl. 1.1 Model neurona

1.1. Prijenosne funkcije

Prijenosna funkcija određuje izlaz neurona temeljeno na ulaznim podacima po formuli (1) [1].

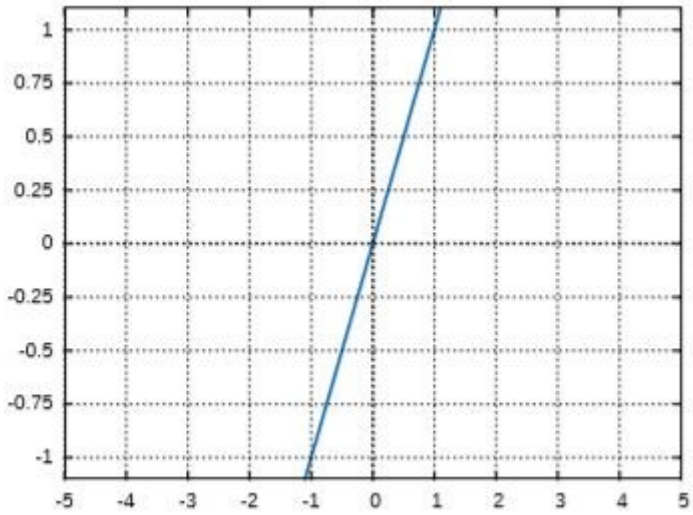
$$y = PrijenosnaFuncija(\sum(težina \cdot ulaz) + pomak) \quad (1)$$

Primjer često korištene nelinearne prijenosne funkcije, funkcija skoka, prikazana je na slici (Sl. 1.2).



Sl. 1.2 Funkcija Skoka

Na slici (Sl. 1.3) vidljiva je linearna funkcija koja će biti korištena u nastavku rada.



Sl. 1.3 Linearna Funkcija

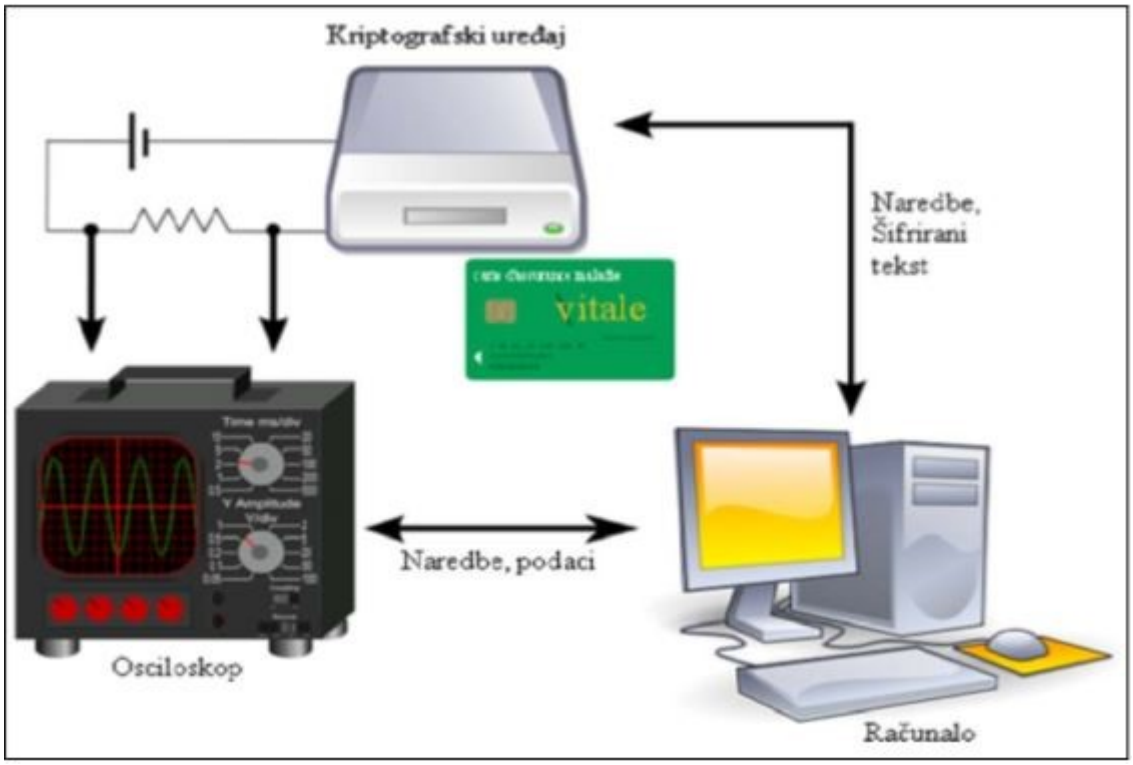
1.2. Težine

Težinama i pragovima neuronske mreže određeno je u koje vrijednosti će se preslikati ulazni uzorak. Težine služe da povećaju ili smanje jačinu signala pri konekciji. Dobro istrenirana neuronska mreža je od velike važnosti, a ono što ju razlikuje od loše istrenirane su upravo težine. Više o neuronskim mreža za ovaj rad nije potrebno te se više informacija može pronaći na [4][5].

2. Napad analizom napajanja

Napad analizom napajanja spada u SCA. Napadi temeljeni na mjerenjima napajanja koriste dva glavna pristupa :

- Jednostavna analiza temelji se na promatranju i analiziranju jednog valnog oblika potrošnje energije računalnog sustava [2].
- Diferencijalna analiza napajanja prikazana na slici (Sl. 2.1), čija ideja je preuzeta iz [6], temelji se na promatranju i analiziranju velikog broja valnih oblika napajanja, a česta metoda koja će se implementirati u ovom radu je korelacijska analiza napajanja [2].



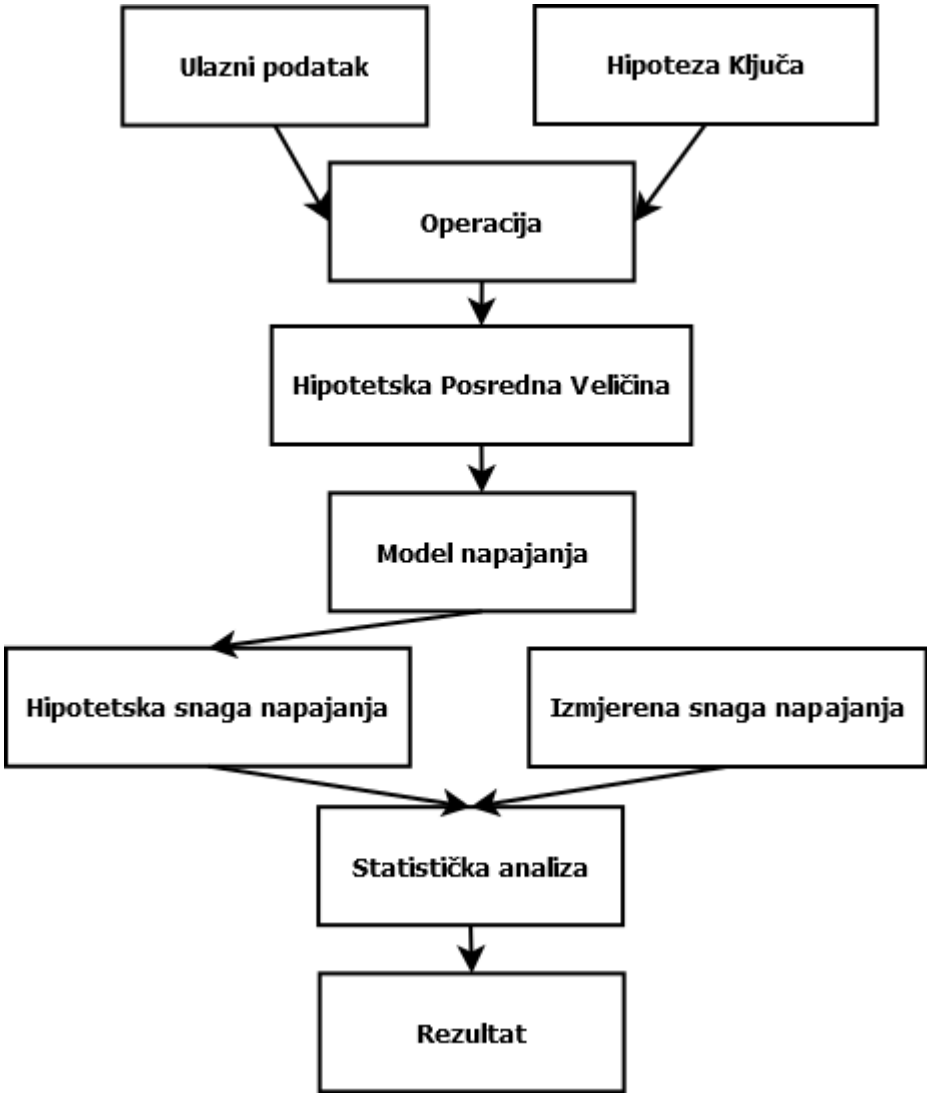
Sl. 2.1 Diferencijalna analiza napajanja

2.1. Korelacijska analiza napajanja

CPA je metoda koja koristi veliki broj mjerenja modela zbog pronalaska tajnog ključa. Tajni ključ se pronalazi statističkim uspoređivanjem hipotetskog modela napajanja s izmjerenim napajanjem sklopa kojeg napadamo. Između tih

podataka pokušavamo pronaći korelaciju, a najčešći način je koristeći Pearsonov koeficijent korelacije [3].

Na slici (Sl. 2.2) je prikazan tijek CPA. Ideja dijagrama preuzeta je iz [2]. Za zadane ulazne podatke i izmjerenu snagu napajanja postupak je sljedeći. Određujemo hipoteze ključa ovisno o traženom tajnom ključu. Pomoću ulaznih podataka i hipoteza ključa dolazimo do posredne vrijednosti koju šaljemo u model napajanja. Model napajanja određuje hipotetsku snagu napajanja koja će u ovom radu biti određena preko Hammingove težine posredne vrijednosti koja je u direktnom odnosu sa snagom napajanja. Konačno uspoređujemo hipotetsku snagu napajanja s izmjerenom i statistički uspoređujemo vrijednosti. Hipoteza s najvećom korelacijom odgovarat će vrijednosti ključa [3][7].



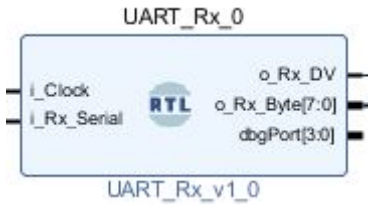
Sl. 2.2 Slijed korelacijske analize napajanja

3. Postavljanje eksperimenta

Cilj eksperimenta je implementacija napada na jednoslojnu neuronsku mrežu te pomoću CPA pokušati izvući parametre težina temeljeno samo na informacijama dobivenim iz fizičke implementacije sustava. Napadač ima fizički pristup neuronskoj mreži i može izmjeriti sve podatke vezane za napajanje sklopa. Također, napadač šalje ulazne podatke što znači da su mu ti podaci poznati. Napadač nema pristup bilo kakvim informacijama vezanima za arhitekturu mreže.

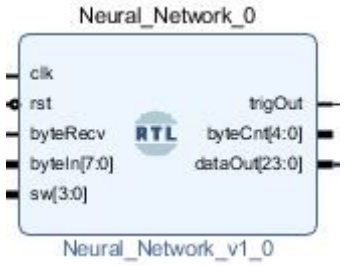
3.1. Projektiranje jednoslojne umjetne neuronske mreže

Jednoslojna umjetna neuronska mreža definirana je u jeziku Verilog, a implementirana na FPGA pločici modela Artix-7 na razvojnoj platformi Arty A7-35T. Glavna zadaća koda je primanje 16 ulaznih podataka težine 8 bitova preko sučelja integriranog asinkronog prijemnika/odašiljača, njihovo množenje s odgovarajućom 8 bitnom težinom i sumiranje. Na slici (Sl. 3.1) je prikazana blok shema integriranog asinkronog prijemnika/odašiljača.



Sl. 3.1 Blok shema integriranog asinkronog prijemnika/odašiljača

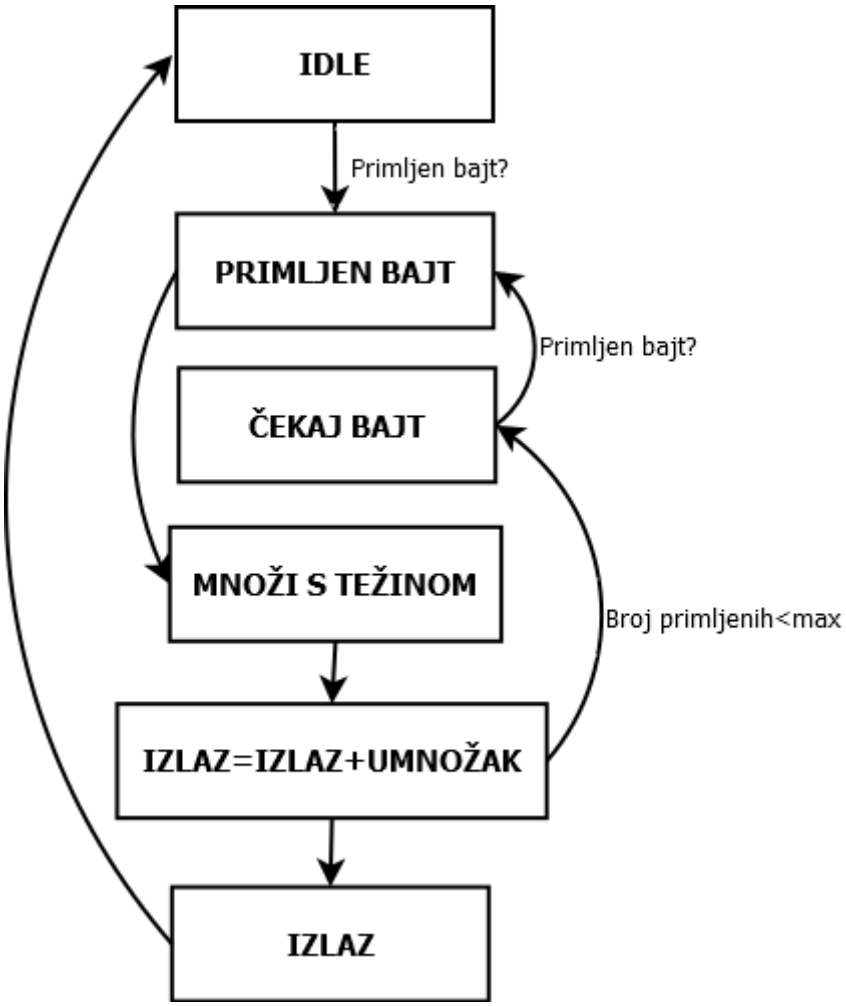
Na ulaz `i_Rx_Serial` dolaze podaci koji se asinkrono prosljeđuju na izlaz `o_Rx_Byte` koji se dalje odašilju na ulaz neuronske mreže. Blok shema neuronske mreže prikazana na slici (Sl. 3.2) prima na ulaz `byte_Recv` signal o primitku bajta, a na ulaz `byteIn` prima odgovarajući bajt.



Sl. 3.2 Blok shema neuronske mreže

Perceptron je izveden kao konačni automat koji se sastoji od konačnog broja stanja. Prijelazi između stanja određeni su duljinom radnog takta, frekvencije podešene na $f = 10$ Mhz, odnosno promjenom signala `clk` iz 0 u 1. Unutar jednog radnog takta izvršava se jedna operacija. Dijagram toka sa slike (Sl. 3.3) opisuje način rada implementiranog perceptrona. U početnom stanju `IDLE` varijable imaju vrijednost 0. Na prvi primljeni bajt trenutno stanje prelazi u stanje `PRIMLJEN BAJT` gdje pratimo broj ulaznih podataka jednog seta, a potom odmah u stanje `MNOŽI S TEŽINOM` gdje se taj ulazni bajt množi s odgovarajućom težinom. Težine su slučajne vrijednosti određene na početku programa koje su jednake za svaki novi set ulaznih podataka. Nakon množenja, umnožak se nadodaje na sumu što predstavlja sumiranje vrijednosti u tijelu umjetnog neurona. Stanje zbrajanja provjerava broj ulaznih podataka seta. Ako je broj jednak maksimalnom broju ulaza jednog seta suma se šalje u prijenosnu funkciju, inače automat prelazi u stanje čekanja sljedećeg bajta. Pošto je prijenosna funkcija linearna, sumu se prosljeđuje na izlaz. Nakon što se obradi jedan set ulaznih podataka automat se vraća u stanje `IDLE` spremno za prihvat novog set podataka.

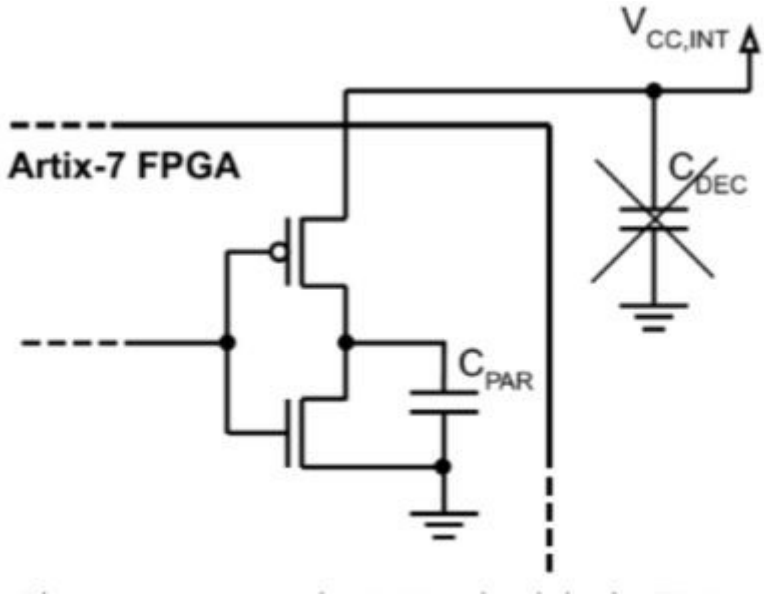
Implementirani perceptron se sastoji od jednog sloja odnosno umjetnog neurona i, iz perspektive napadača, skrivenog seta težina kojeg pokušava izvući. Operacija množenja ulaznog podatka s težinom osigurava „curenje“ informacija o naponu perturbacija na naponu napajanja. Ta količina je proporcionalna Hammingovoj težini umnoška [7].



Sl. 3.3 Dijagram toga implementiranog perceptrona

3.2. Metodologija za mjerenje napona FPGA sklopa

Na slici (Sl. 3.4) preuzetoj iz [3] je primjer CMOS invertera sadržanog unutar Artix-7 FPGA pločice. On prikazuje način prikupljanja informacija pomoću izvora napajanja $V_{CC,INT}$. Kod invertiranja vrijednosti izlaza bilo koje operacije dolazi do punjenja i pražnjenja parazitskog kapaciteta C_{PAR}, pri čemu nastaju smetnje na napajanju $V_{CC,INT}$. Kapaciteti C_{DEC}, koji su namijenjeni uklanjanju tih smetnji, su otklonjeni da bi se spomenute perturbacije mogle izmjeriti [3]. Mjerenja su izvršena pomoću osciloskopa *Yokogawa DLM403*. Mjerenja započinju okidanjem signala `trigOut`.



Sl. 3.4 CMOS sklop unutar Artix-7 FPGA pločice

3.3. Pearsonov koeficijent korelacije

Formula preuzeta s [3] računa koeficijent korelacije između Hammingove težine rezultata množenja H i izmjerene snage napajanja V je:

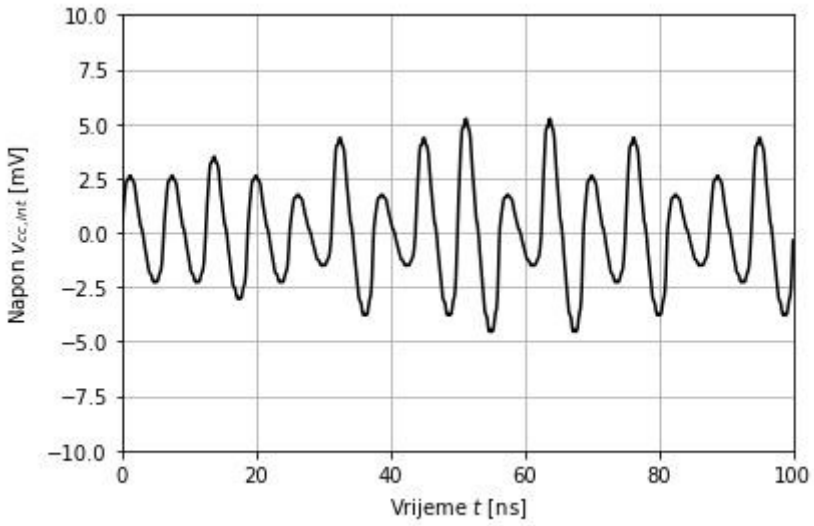
$$r_{H,V} = \frac{\sum_{i=1}^{n}\left(h_i - \overline{h}\right)\left(v_i - \overline{v}\right)}{\sqrt{\sum_{i=1}^{n}\left(h_i - \overline{h}\right)^2}\sqrt{\sum_{i=1}^{n}\left(v_i - \overline{v}\right)^2}} \qquad (2)$$

gdje n predstavlja broj mjerenja, h i v Hammingovu vrijednost odnosno napajanje primjera, a $\overline{h}$ i $\overline{v}$ njihovu srednju vrijednost. Pearsonov koeficijent korelacije kreće se od +1 što pokazuje savršenu pozitivnu korelaciju do –1 što pokazuje savršenu negativnu korelaciju [3]. Ako koeficijent iznosi 0 to znači da nema linearne korelacije, ali ne znači da se varijable nezavisne.

4. Reverzno inženjerstvo neuronske mreže

4.1. Mjerenje napona napajanja

Na ulaz neuronske mreže se šalje $N = 256$ setova ulaznih podataka. Svaki set podataka sastoji se od 16 ulaznih poruka svaka težine od 8 bita. Jedan set predstavlja set ulaznih podataka neuronske mreže implementirane na FPGA. Osciloskopom se izmjeri N valnih oblika od kojih se svaki oblik sastoji od 16 perioda. Na slici (Sl. 4.1) se vidi primjer valnog oblika s različitim vršnim vrijednostima. Svaki period valnog oblika predstavlja jedno množenje neuronske mreže odnosno množenje ulaza $x_i, i \in [0,15]$ s odgovarajućom težinom $w_i, i \in [0,15]$. Ulazni podaci su 8 bitni brojevi Hammingove težine u rasponu [0 - 8].



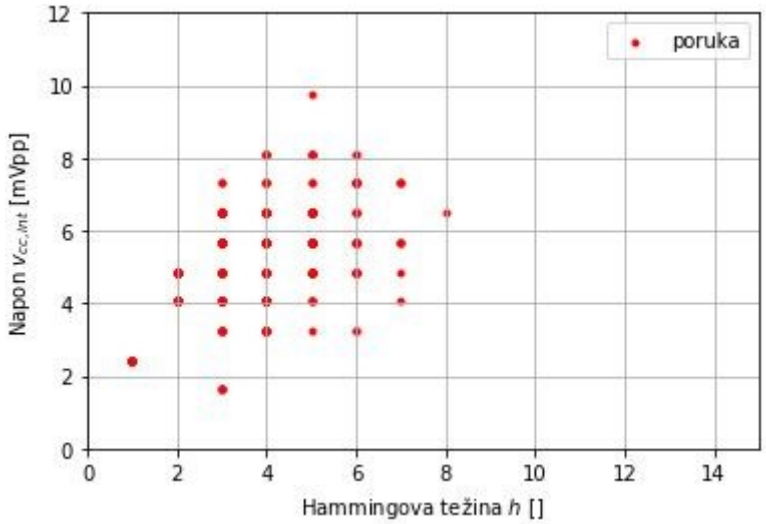
Sl. 4.1 Primjer valnog oblika snimljenog osciloskopom

4.2. Projektiranje programske podrške za obradu mjerenja

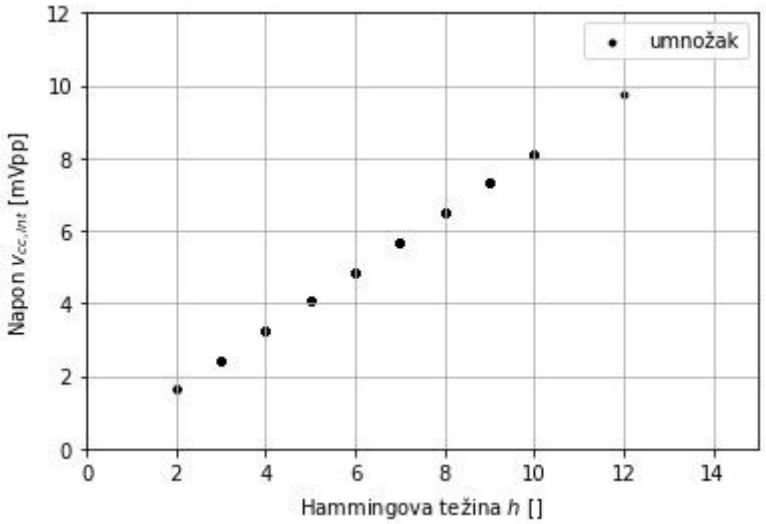
Skripta za obradu mjerenja napajanja i izvlačenje parametara korelacijskom analizom napisana je u jeziku Python. Prije korelacijske analize potrebno je pronaći maksimalne vrijednosti perioda (engl. *peaks*) za svaku težinu pojedinačno. Prvo su pronađene maksimalne vrijednosti za svaki period valnog oblika, a to je određeno Python – ovom gotovom funkcijom iz knjižnice (eng. library) scipy.signal :

```
arrayRazlikaVršnihVrijednosti=scipy.signal.find_peaks(valni_oblik)
```

Maksimalne vrijednosti su tada grupirane po periodima odnosno za svaki period $i, i \in [0,15]$ napravljena je lista gdje su na indeks i dodane vršne vrijednosti pripadnog perioda napajanja za N izmjerenih valnih oblika. Na slici (Sl. 4.2) su maksimalne vrijednosti napajanja prikazane u odnosu na Hammingove vrijednosti ulaznih podataka za x_0. Sl. 4.2 pokazuje da ne postoji korelacija između maksimalnih vrijednosti i težina ulaznih podatke. Slika (Sl. 4.3) prikazuje odnos maksimalnih vrijednosti napajanja i Hammingovih težina umnožaka ulaznih podataka i pripadnih točnih težina za x_0, w_0. Ovi rezultati pokazuju jasnu korelaciju između Hammingove težine umnoška i maksimalnih vrijednosti napona napajanja.



Sl. 4.2 Maksimalne vrijednosti napajanja s obzirom na Hammingove težina ulaznih vrijednosti



Sl. 4.3 Maksimalne vrijednosti napajanja s obzirom na Hammingove težina umnoška

4.3. Izvlačenje tajnih težina neuronske mreže

U prošlom poglavlju ustanovljena je korelacija između maksimalnih vrijednosti napajanja i Hammingovih težina umnožaka ulaza i pripadnih točnih težina. Skrivene težine koje su implementirane i koje je cilj izvući su redom zadane:

$$w = \{164, 74, 204, 28, 10, 179, 144, 94, 237, 80, 190, 112, 59, 40, 224, 233\}$$

ili u binarnom obliku :

$$w = \{10100100, 01001010, 11001100, 00011100, 00001010, 10110011, 10010000,$$
$$01011110, 11101101, 01010000, 10111110, 01110000, 00111011, 00101000,$$
$$11100000, 11101001\}.$$

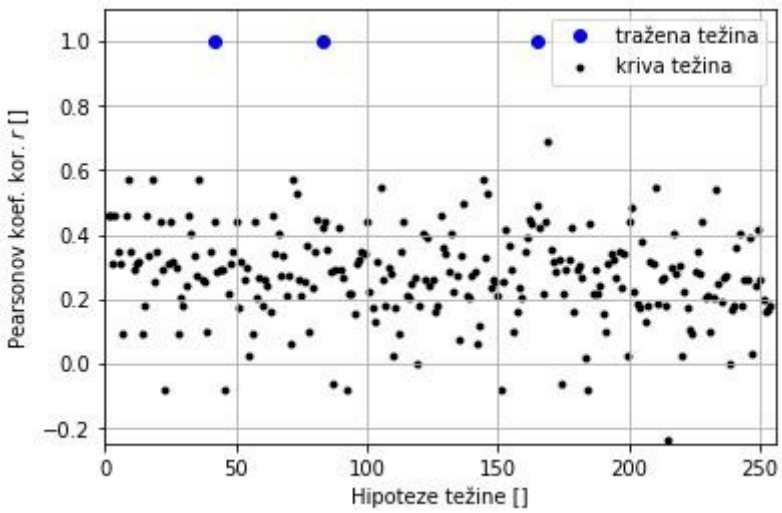
Težine se šalju kao 8 bitni brojevi što određuje 256 hipoteza po težini. Iterirajući po ulazima skripta uzima svaku hipotezu ključa te ju množi s poljem ulaznih podataka tog ulaza. Hammingovu težinu svakog umnoška računa funkcija:

```
def HammingovaTežina(umnožak):
      return bin(umnožak).count("1")
```

Pearsonov koeficijent korelacije će odrediti jačinu korelacije između maksimalnih vrijednosti napajanja i polja Hammingovih težina umnožaka ranije izračunatih. Za računanje Pearsonov – og koeficijenta korelacije korištena je gotova Python – ova funkcija iz knjižnice numpy :

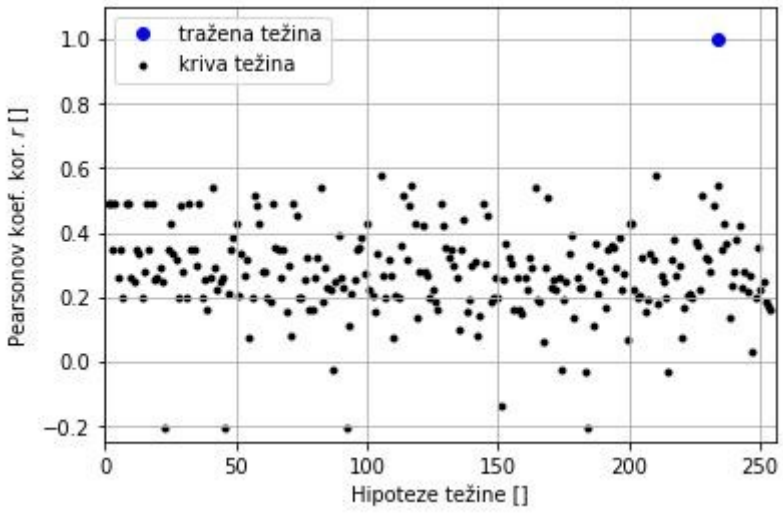
```
numpy.corrcoef(HTlista, MAKSlista)
```

gdje `HTlista` predstavlja listu težina umnožaka, a `MAKSlista` listu maksimalnih vrijednosti. Slika (Sl. 4.4) prikazuje Pearsonove koeficijente korelacije s obzirom na hipoteze prve težine. Iz Sl. 4.4 se vidi da minimalna korelacija iznosi -0.23345, a maksimalna 0.9999 što govori da u slučaju 1. težine dolazi do skoro maksimalne povezanosti. Vrijednosti koje odgovaraju maksimalnom koeficijentu korelacije su redom 41, 82 i 164 odnosno 00101001, 01010010 i 10100100. Primjećuje se dobivanje točne vrijednosti za 1. težinu odnosno 164, ali i dobivanje 2 dodatna kandidata koja se dobiju dijeljenjem točne vrijednosti s 2 ili pomicanjem binarnog broj 1 mjesto u desno (engl. shift). Počevši od jedne vrijednosti nove se mogu dobiti dijeljenjem s 2 dok je god broj djeljiv ili množenjem s 2 dok god umnožak ne pređe raspon hipoteza.



Sl. 4.4 Pearsonov koeficijent korelacije u odnosu na hipoteze težina

Graf sa slike (Sl. 4.5) prikazuje napad na posljednju 15. težinu. Primjećuje se da maksimalni koeficijent korelacije iznosi 1.0 i da pripada hipotezi 233 odnosno 11101001. Shodno prethodnom zaključku 8 bitni broj sa znamenkom 1 na kraju, tj. neparni 8 bitni broj, nije djeljiv s 2. Također 8 bitni broj koji sadrži 1 na najvišem bitu se ne može množiti s 2, a da ne pređe raspon hipoteza. Stoga je to najbolji slučaj CPA analize jer se dobije samo 1 kandidat za tajnu težinu neuronske mreže. U tablici (Tablica 1) se nalaze konačni rezultati CPA analize.



Sl. 4.5 Primjer najboljeg slučaja CPA analize s 1 mogućom težinom

Tablica 1 Dobivene težine

1. težina	2. težina	3. težina	4. težina	5. težina	6. težina
00101001	00100101	00110011	00000111	00000101	10110011
01010010	01001010	01100110	00001110	00001010	
10100100	10010100	11001100	00011100	00010100	
			00111000	00101000	
			01110000	01010000	
			11100000	10100000	
7. težina	**8. težina**	**9. težina**	**10. težina**	**11. težina**	**12. težina**
00001001	00101111	11101101	00000101	01011111	00000111
00010010	01011110		00001010	10111110	00001110
00100100	10111100		00010100		00011100
01001000			00101000		00111000
10010000			01010000		01110000
			10100000		11100000
13. težina	**14. težina**	**15. težina**	**16. težina**		
00111011	00000101	00000111	11101001		
01110110	00001010	00001110			
11101100	00010100	00011100			
11101001	00101000	00111000			
	01010000	01110000			
	10100000	11100000			

Zaključak

Umjetne neuronske mreže su vrlo raširen oblik algoritama zbog svoje prilagodljivosti, učinkovitosti i praktičnosti. Njihova vrijednost i efikasnost raste ovisno o arhitekturi i utreniranosti mreže, a za tu su najbitnije težine neuronske mreže. Ovaj rad je pokazao način reverznog inženjerstva neuronske mreže i izvlačenja težina promatrajući samo propuste fizikalne implementacije na FPGA pločicu.

Prvo je ustanovljeno da se, nakon uklanjanja izvana kapaciteta zaduženih za uklanjanje perturbacija na napajanju, mogu izmjeriti smetnje koje daju uvid u rezultate operacija. Nakon izvršenih mjerenja ustanovljena je korelacija između maksimalnih vrijednosti napajanja i rezultata množenja. Koristeći Pearsonov koeficijent korelacije izvršena je CPA nad izmjerenim podacima.

CPA je pronašla sve kandidate potencijalnih skrivenih težina. Vidljivo je da je CPA otkrila svaku točnu težinu kao i još neke ostale kandidate. CPA je izvukla sve ključeve koji su djeljivi s 2^n ili množenjem s 2^n ne prelaze mogući raspon ključeva. Obje radnje su pomicanje binarnog broj u desno ili lijevo n puta. Te radnje imaju jednak efekt i na rezultat množenja s težinom. Rezultat je pomaknut n puta u desno ili lijevo, a njegova Hammingova težina se neće promijeniti i zato te ključeve ne možemo razlučiti.

CPA je vrlo korisna jer bitno smanji broj kandidata za točan ključ. U najboljem slučaju se dobije jedan kandidat – u ovom radu to su svi neparni ključevi veći od 128 (2^{N-1}). U najgorem slučaju daje N kandidata – u ovom radu to bi bile težine 1, 2, 4, 8, 16, 32, 64, 128 (0000001, 00000010, ..., 10000000). U najgorem slučaju suzi se broj kandidata 32 puta za 8 bitni tajni ključ što nam pokazuje veliku efikasnost CPA.

Valja primijetiti da je sklopovska podrška digitalnog sustava jako bitna za zaštitu vlastite implementacije. Mogućnost ovakvog načina krađe parametara neuronske mreže je kritičan te su poboljšanja sigurnosti u tom smjeru prijeko potrebna.

Literatura

[1] Batina, L., Bhasin, S., Jap, D., Picek, S. *{CSI}{NN}: Reverse Engineering of Neural Network Architectures Through Electromagnetic Side Channel*. 28th {USENIX} Security Symposium ({USENIX} Security 19), 2019, pp. 515-532.

[2] Mangard, S., Oswald, E., Popp, T. *Power analysis attacks: Revealing the secrets of smart cards,* Springer Science & Business Media. vol. 31, 2008.

[3] Kundrata, J., Fujimoto, D., Hayashi, Y., Barić, A. *Comparison of Pearson correlation coefficient and distance correlation in Correlation Power Analysis on Digital Multiplier.* 2020.

[4] Bašić, B. D., Čupić, M., Šnajder, J. *Umjetne neuronske mreže*. Zagreb, 2008.

[5] Dumančić., S. *Neuronske mreže*. Diplomski rad. Sveučilište Josipa Jurja Strossmayera u Osijeku, Osijek, 2014.

[6] Guillemet, C., San Pedro, M., Servant, V. *Side-Channel assessment of Open Source Hardware Wallets*, SSTIC, Rennes, 2019.

[7] Brier, E., Clavier, C., Olivier F. *Correlation power analysis with a leakage mode*l. International Workshop on Cryptographic Hardware and Embedded Systems, Springer, Berlin, Heidelberg, 2004.

Sažetak

Implementacija napada na neuronsku mrežu analizom napajanja FPGA sklopa

Digitalni sustavi mogu biti sigurnosno ranjivi na razini programske i sklopovske podrške. Napadi koji koriste ranjivosti sklopovske podrške su tzv. side – channel napadi. U ovom radu na FPGA pločicu je implementirana jednoslojna neuronska mreža s linearnom funkcijom. Izvedeno je mjerenje napona napajanja FPGA sklopa. Napisana je programska podrška za obradu rezultata mjerenja te za njihovo daljnje korištenje. Izvršena je korelacijska analiza napajanja te su korištenjem koeficijenta korelacije pronađeni tajni ključevi, odnosno težine, neuronske mreže. Tim korakom je završen postupak reverznog inženjerstva neuronske mreže.

Ključne riječi

Neuronske mreže, side – channel napadi, FPGA, napad analizom napajanja, korelacijska analiza napajanja, Pearsonov koeficijent korelacije, Hammingova težina, reverzno inženjerstvo

Summary

Implementation of a power supply side-channel attack on FPGA-based neural network

Digital systems can be vulnerable at the software and hardware level. Attacks that exploit hardware support vulnerabilities are side - channel attacks. In this paper, a single-layer neural network with a linear function is implemented on an FPGA board. FPGA supply voltage measurement was performed. Software has been written for processing measurement results and their further use. A power correlation analysis was performed and secret keys, ie weights, of the neural network were found using the correlation coefficient. This step completed the process of reverse neural network engineering.

Key words

Neural networks, side-channel attack, FPGA, power analysis, correlation power analysis, Pearson correlation coefficient, Hamming weight, reverse engineering

Skraćenice

CPA	*Correlation Power Analysis*	korelacijski analiza napajanja
SCA	*Side-Channel Attack*	