

Automatizovano simulaciono okruženje za ispitivanje inteligentnih senzora zasnovanih na MEMS

Aleksandar Tucakov, Nebojša Pjevalica, Zoltan Pele

Sadržaj — Rad prikazuje jedan pristup u projektovanju i korišćenju automatizovanog okruženja za ispitivanje savremenih inteligentnih senzora zasnovanih na mikroelektromehaničkim (MEMS) davačima, koji poseduju digitalni sprežni modul.

Ključne reči — Automatizacija ispitivanja, ispitivanje integriranih kola, MEMS, verifikacija.

I. UVOD

ISPITIVANJE spada među najznačajnije faze u procesu razvoja integriranih kola. U toku ispitivanja proverava se funkcionalnost razvijenog modela integriranog kola pre njegovog puštanja u komercijalnu proizvodnju. Za opis modela integriranog kola koriste se jezici za opis fizičke arhitekture kao što su Verilog ili VHDL [1], [2], [3] dok se samo ispitivanje/simulacija obavlja specijalizovanim softverskim okruženjima kao što su Modelsim i NCSim.

Blagovremeno ispitivanje pojedinih razvijenih modula (podblokova) povećava konačnu pouzdanost integriranog kola. Nakon verifikacije svih pojedinačnih modula, neophodno je objediniti ih u celinu koja mora odgovarati željenom integriranom kolu. Ovako formiran objedinjen model pruža mogućnost verodostojne provere sistema pre ulaska u proces serijske proizvodnje, nakon koga naknadne ispravke više nisu moguće.

Automatizacija procesa ispitivanja ima mnoge prednosti u odnosu na ručno ispitivanje. Pre svega ljudsko vreme je skupo a takođe, veoma je teško zadržati koncentraciju u dužem vremenskom periodu, pa je i verovatnoća da će čovek pogrešiti veća. Iako je za projektovanje automatizovanog okruženja u početku potrebno izdvojiti malo više truda, taj trud kasnije rezultuje većom pouzdanošću, te uštedom vremena i novca.

Pri ispitivanju je korišćen HDL model koji predstavlja

Ovaj rad je delimično finansiran od Ministarstva za nauku Republike Srbije, projekat 11005, od 2008. god.

A. Tucakov, Fakultet tehničkih nauka, Novi Sad, Srbija, (telefon: +381-(0)21-4801-186; faks: +381-(0)21-4801-196; e-mail: aleksandar.tucakov@rt-rk.com)

N. Pjevalica, Fakultet tehničkih nauka, Novi Sad, Srbija, (e-mail: nebojsa.pjevalica@rt-rk.com)

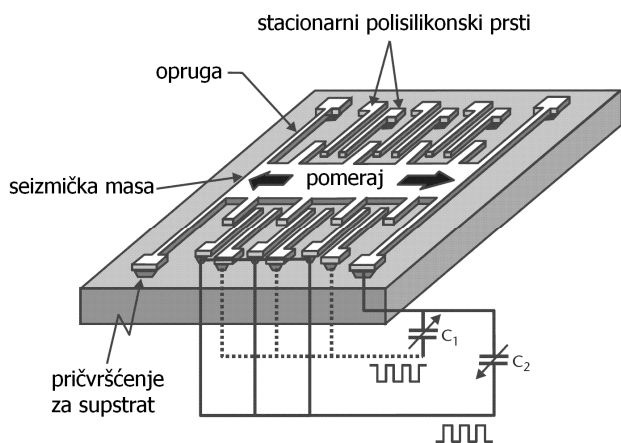
Z. Pele, Fakultet tehničkih nauka, Novi Sad, Srbija, (e-mail: zoltan.pele@rt-rk.com)

deo senzora bez MEMS-a. Za sintezu je korišćen Xilinx ISE 9.2 [4] a za simulaciju softverski paket NCSim [5] kompanije Cadence.

II. KONCEPT SENZORA ZASNOVANIH NA MEMS TEHNOLOGIJI

U mikroelektromehaničke sisteme (MEMS) spadaju uređaji dimenzija između 1 μ m i 1mm proizvedeni kombinovanjem mehaničkih i električnih struktura [6]. Proizvodna tehnologija najvećim delom preuzeta je od proizvođača integriranih kola. Značajno smanjene dimenzije ovakvih uređaja opredeljuju njihove mehaničke osobine, izuzetno mala masa, smanjen moment inercije, nizak koeficijent trenja, smanjeno mehaničko habanje doprinose dugotrajnosti. Navedene karakteristike implicirale su vrlo širok spektar primena pogotovo u senzoricima. Tipični predstavnici MEMS senzora su akcelerometri, žiroskopi, senzori pritiska, protoka, termometri, anemometri, itd.

U osnovi MEMS senzor pretvara mehaničku ili neku drugu fizičku veličinu u električnu, koja se digitalizuje i prilagođava korisniku (filtrira, reskalira, itd.). U konkretnom slučaju je analiziran i simuliran akcelerometar koji meri ubrzanje po sve tri ose Dekartovog koordinatnog sistema. Akcelerometar se sastoji od seizmičke mase koja pretvara ubrzanje u silu, čijim delovanjem seizmička masa menja svoj položaj [7]. Njeno ogibljenje je elastično tako da je pomeraj srazmeran sili i menja kapacitet kondenzatora kojeg formira pokretni deo senzora sa „stacionarnim polisilikonskim prstima”. Ovakva struktura je predstavljena na Sl. 1.



Sl. 1. Akcelerometrički senzor zasnovan na pretvaranju ubrzanja u pomerač.

Senzor u sebi sadrži tri ovakve strukture, orijentisane kolinearno sa svakom od osa koordinatnog sistema. Signali sa kondenzatora se vode na A/D pretvarač gde se pretvaraju u numeričke vrednosti srazmerne ubrzanju svake od osa. Analiza, simulacija i ispitivanje MEMS elementa izlazi iz okvira ovog rada. Realni podaci o ubrzanju preuzeti su sa fizičkog senzora i memorisani u ispitne datoteke. Ovako formiran skup ispitnih datoteka korišćen je pri ispitivanju digitalnog sklopa razvijanog senzora.

III. OPIS ISPITNOG OKRUŽENJA

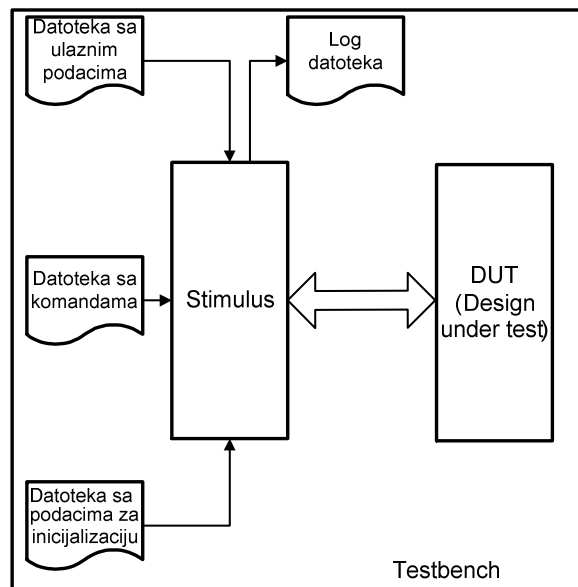
Kao što je već napomenuto ispitivanje je vršeno nad VHDL dizajnom koji predstavlja deo senzora bez MEMS-a. Zbog nepostojanja adekvatnog modela MEMS-a realizovanog u VHDL-u javila se potreba da se na neki način simulira funkcionalnost samog MEMS-a. Ovo je učinjeno na taj način što je formirana datoteka sa ispitnim vektorima čije vrednosti predstavljaju vrednosti ubrzanja na izlazu samog MEMS-a. Na taj način iščitavanjem vrednosti iz datoteke i njihovim slanjem ka senzoru sa određenom frekvencijom simulira se funkcionalnost MEMS-a u celokupnom sistemu. Sva komunikacija prema senzoru obavlja se putem SPI (Serial Peripheral Interface) komunikacije.

Za ispitivanje je korišćen specijalizovan softverski paket namenjen za softversku simulaciju NCSim. Softverski paket NCSim sastoji se od sledećih komponenti:

- Simulator za Verilog jezik – NCVlog
- Simulator za VHDL jezik – NCVHDL
- Grafičko okruženje za rad sa NCVHDL i NCVlog – SimVision

Da bi se ispitao neki HDL model neophodno je pre svega napisati HDL kod koji će se ponašati kao "spoljašnje okruženje" za ispitivani HDL model. Ovakav HDL kod se na engleskom naziva Testbench [8] dok se modul koji se ispituje naziva DUT (Design under test). Uloga Testbench-a je da generiše takt signale (clock), signale za dovođenje

DUT-a u početno stanje (reset) i neophodne signale za pobuđivanje ispitivanog HDL modela. Isto tako Testbench posmatra izlazne signale ispitivanog HDL modela i u zavisnosti od njih generiše dodatne signale za pobuđivanje DUT-a ili ih beleži u jednu log datoteku kako bi korisnik kasnije bio u mogućnosti da ih pregleda.



Sl. 2. Izgled ispitnog okruženja

Na Sl. 2 je prikazan izgled ispitnog okruženja. Na samom početku neophodno je izvršiti inicijalizaciju samog modela koji se ispituje. Ovo se obavlja tako što se iz datoteke pročitaju podaci koji služe za inicijalizaciju i zatim se proslede ka modelu koji se ispituje. Nakon toga započinje se sa slanjem vrednosti ubrzanja ka modelu koji testiramo. Vrednosti ubrzanja predstavljaju izlaze MEMS-a koji postoji u realnom senzoru. U pitanju su osmobicne vrednosti ubrzanja predstavljene u komplementu dvojke. Ove vrednosti prosleđuju se ka modelu senzora određenom frekvencijom koju korisnik može unapred definisati pre početka ispitivanja.

IV. PROCES ISPITIVANJA

Pri ispitivanju su korišćene tri ulazne tekstualne datoteke sa podacima. Prva datoteka sadrži podatke potrebne za inicijalizaciju modela koji se ispituje. Svaka linija ove datoteke predstavlja jedan podatak za inicijalizaciju i sastoji se iz adrese internog registra senzora u koji treba upisati vrednost i vrednosti koja se upisuje. Adrese i vrednosti prikazane su u heksadecimalnom formatu iz opsega 00h - FFh. Podaci iz ove datoteke prosleđuju se ka ispitivanom modelu na samom početku simulacije čime je sistem inicijalizovan i spreman za ispitivanje njegove funkcionalnosti.

Druga datoteka sadrži realne podatke o ubrzanju koji su preuzeti sa fizičkog senzora i zatim memorisani u datoteku. Uloga ove datoteke je da simulira funkcionalnost MEMS elementa tako što se iz nje iščitaju vrednosti koje se zatim

prosleđuju ka ispitivanom modelu. Frekvencija kojom se vrednosti ubrzanja prosleđuju ka ispitivanom modelu može se podesiti pre početka same simulacije ili se može koristiti podrazumevana vrednost. Za svaki ispitivani slučaj potrebno je formirati posebnu datoteku ovog tipa, u zavisnosti od toga koji modul (podblok) i koja funkcionalnost se želi ispitati.

Datoteka sa komandama sadrži komande koje su potrebne za ispitivanje senzora. Uloga ove datoteke je da Stimulusu obezbedi podatke neophodne za sukcesivno pobuđivanje modula (podblokova) ispitivanog modela. Nju kreira sam korisnik, u zavisnosti od toga koji modul i koje funkcionalnosti se žele ispitati. Svaka linija ove datoteke predstavlja jednu komandu za čitanje ili upis u željeni registar senzora. Izgled dela datoteke sa komandama je prikazan na Sl. 3.

```

1 1000 34 0 AB
2 1000 0d 0 47
3 1000 0e 0 01
4 1000 27 1 00
5 1000 28 1 00
6 1000 08 0 d7
7 1000 10 0 05
8 2000 1a 0 55

```

Sl. 3. Izgled dela datoteke sa komandama

Linija datoteke sadrži trenutak u kojem treba izvršiti datu komandu, predstavljeno vremenskim intervalom od početka simulacije u mikrosekundama, adresu internog registra senzora (heksadecimalno) u koji treba upisati ili iz kojeg treba iščitati vrednost, indikator da li treba izvršiti upis (indikator je tada 0) ili čitanje (indikator je tada 1) i vrednost koju treba upisati (heksadecimalno). U slučaju komande za čitanje iz internog registra senzora vrednost u poslednjoj koloni komandne linije nije od interesa, međutim neophodno je upisati proizvoljnu vrednost iz opsega 00h – FFh zbog načina rada funkcije za čitanja iz datoteke.

Nakon formiranja ove tri vrste ulaznih datoteka i izbora izlaza koji se žele beležiti u izlaznu log datoteku započinje se sa procesom ispitivanja (simulacije). Kao rezultat simulacije koja traje određeni vremenski period tokom koga prisustvo čoveka nije neophodno dobijene su izlazne log datoteke, za svaki ispitivani slučaj po jedna, koje su zatim detaljno proučavane kako bi se izvršilo poređenje aktuelnog ponašanja digitalnog sistema sa željenim i kako bi se otkrile eventualne greške u digitalnom sistemu.

V. REZULTATI ISPITIVANJA

Detaljno ispitivanje ponašanja digitalnog senzorskog podbloka u okruženju opisanom u poglavlju III omogućilo je brzu lokalizaciju grešaka i njihovo ispravljanje. Za proveru svakog elementa digitalnog sklopa formirana je posebna datoteka sa pobudama. Poređenjem aktuelnog ponašanja sistema sa željenim, detekcija grešaka postaje efikasna i fokusirana.

Nakon ispravljanja uočenih grešaka (konceptijskih i slučajnih) finalni dizajn je verifikovan u konačnom testu,

gde su sukcesivno aktivirane sve pobude jedna za drugom uz praćenje ponašanja sistema.

VI. ZAKLJUČAK

Imajući u vidu visoku cenu proizvodnje integrisanog kola, neotkrivene greške mogu proizvod učiniti neupotrebljivim na tržištu. Ovakav gubitak u oštroj tržišnoj konkurenciji, kakva je danas prisutna u svim oblastima informacione tehnologije, mogao bi biti nenadoknadiv.

Korak dalje u pristupu predprodukcijском ispitivanju bio bi uključivanje MEMS elementa u simulacioni model. Ovakav sveobuhvatni pristup omogućio bi gotovo potpuno verodostojan proces verifikacije svih segmenata integrisanog kola u jedinstvenoj simulaciji. Time bi se dale ispraviti i eventualne greške u samom mikroelektromehaničkom bloku, kao i u sprezi između njega, A/D bloka i digitalnog bloka.

LITERATURA

- [1] K. Skahill, "VHDL for Programmable Logic", Cypress Semiconductor, Addison-Wesley 1996, Menlo Park, CA
- [2] P. Chu, "RTL Hardware Design Using VHDL", Wiley – Interscience, 2006
- [3] S. Brown, Z. Vranesic, "Fundamentals of Digital Logic with VHDL Design", McGraw-Hill, 2000. Blacklick, OH
- [4] Xilinx ISE 9.2 Software manuals, Xilinx Inc., 2007
- [5] Cadence NC-VHDL Simulator help, 2004
- [6] M. Gad-el-Hak, "MEMS Applications", 2nd ed, Boca Raton: CRC Taylor and Francis, 2006.
- [7] N. Maluf, K. Williams, "An Introduction to Microelectromechanical Systems Engineering", Norwood, MA: Artech House, 2004.
- [8] R. Munden, "ASIC and FPGA Verification: A Guide to Component Modeling", Morgan Kaufmann, 2004

ABSTRACT

This paper presents single approach in design and usage of automated environment for MEMS based intelligent sensor testing. The target testing subblock is digital interface.

AUTOMATED SIMULATION ENVIRONMENT FOR INTELLIGENT MEMS BASED SENSORS TESTING

Aleksandar Tucakov, Nebojsa Pjevalica, Zoltan Pele