

Implementacija digitalnih elektronskih kola primenom boundary scan standarda

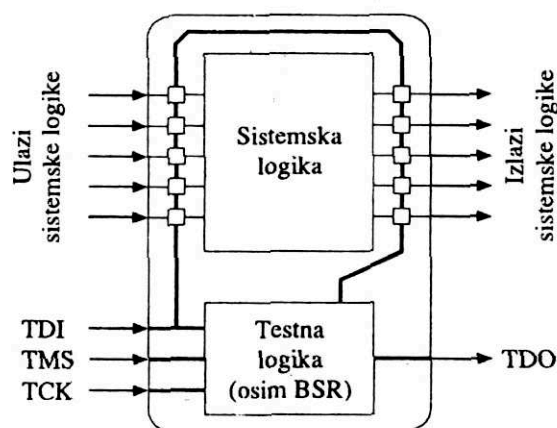
DEJAN M. MAKSIMOVIĆ, VLADAN M. PANIĆ, KRISTIJAN M. ZARKOVIĆ, ZORAN M. PETKOVIC, DRAGIŠA P. MILOVANOVIĆ, DRAGAN TOPISIROVIĆ, VANČO B. LI TOVSKI, Univerzitet u Nišu, Elektronski fakultet, Niš

Stručni rad
UDC:681.327:621.377.621.63=861

Boundary scan koncept projektovanja za testabilnost podrazumeva ugradnju dodatne testne logike u čip ili sistem na štampanoj ploči radi pojednostavljenja testiranja čipa nakon proizvodnje i dijagnostike defekata u toku eksploatacije i održavanja. U ovom radu izložena je jedna praktična implementacija boundary scan standarda za testiranje digitalnih sistema. Dodatna testna logika je projektovana, simulacijom verifikovana, realizovana od diskretnih komponenti na štampanoj ploči i pokazano je da se ona efikasno može upotrebiti za testiranje defekata čiji je efekat permanentno stanje na vezi.

1. UVOD

Danas je boundary scan koncept projektovanja za testabilnost široko prihvaćen od strane gotovo svih proizvođača, kako integrisanih kola, tako i štampanih ploča. Boundary scan koncept garantuje sistematski pristup ugrađenoj logici na čipu ili štampanoj ploči radi testiranja nakon proizvodnje, u toku eksploatacije i prilikom održavanja sistema [1, 2, 3]. Uštede u procesu testiranja koje donosi boundary scan čine ovaj pristup isplativim čak i kod malih serija štampanih ploča [4].



Sl. 1 - Struktura čipa realizovanog u skladu sa BS1149.1

Adresa autora: Dr Dejan Maksimović, Elektronski fakultet, Niš, Beogradska 14

Rad primljen: 27.07.2001.

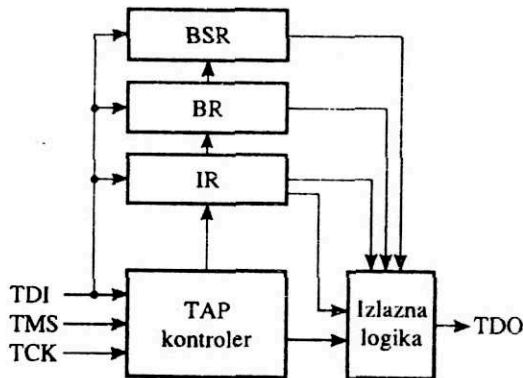
Boundary scan metod projektovanja definisan je IEEE standardom 1149.1 (u daljem tekstu: BS1149.1). Ovaj standard podrazumeva ugradnju dodatne testne logike u čip radi pojednostavljenja testiranja logike na čipu i u njegovom okruženju [5, 6]. Čip realizovan u skladu sa BS1149.1 sastoji se od *sistemske logike* koja obavlja primarnu funkciju čipa i dodatne *testne logike*, kao što je to prikazano na sl. 1. U ovom radu opišaćemo jednu praktičnu implementaciju BS1149.1 standarda.

2. PROJEKTOVANJE TESTNE LOGIKE

Osnovni elementi testne logike su memorijski elementi raspoređeni po obodu integrisanog kola - boundary scan ćelije (BSC). Da bi dospelo do sistemske logike signal sa primarnog ulaza mora da prođe kroz jednu BSC. Takođe, da bi dospelo do primarnog izlaza, signal sa izlaza sistemske logike mora da prođe kroz jednu BSC. Sve boundary scan ćelije povezane su u boundary scan registar (BSR) koji ima mogućnost serijskog i paralelnog upisa.

Osim boundary scan registra, testna logika sadrži još nekoliko celina, kao što je to prikazano na sl. 2. Najvažnije su: bypass registar (BR), instrukcijski registar (IR), TAP (test access port) kontroler i izlazna logika. Pomoću BR-a čip se premošćuje pri testiranju složenijih sistema koji se sastoje od više čipova projektovanih u skladu sa BS1149.1. Učitavanjem odgovarajuće instrukcije u IR bira se jedan od više mogućih testnih režima. TAP kontroler upravlja radom celokupne testne logike. Testna logika je sekvencijalna i taktuje se testnim taktim signalom TCK (test clock).

Kontrola načina rada TAP kontrolera ostvaruje se dovođenjem odgovarajućeg talasnog oblika na TMS (test mode select) ulaz. Spolja gledano, čip projektovan prema BS1149.1 ima četiri dodatna pina: TDI - ulaz za testne podatke, TDO - izlaz za testne podatke, TMS - ulaz za izbor načina rada i TCK - testni takt. Ovi pinovi, posmatrani zajedno, formiraju port za testiranje - TAP (Test Access Port). Izlazna logika predstavlja multiplexer koji odlučuje koji će registar biti vezan između TDI i TDO: BSR, BR ili IR.



Sl. 2 - Blok šema testne logike

TAP kontroler generiše taktne i upravljačke signale za BSR, BR, IR i izlaznu logiku, a realizuje se u obliku konačnog automata sa 16 stanja. Stanja TAP kontrolera opisuju se pomoću četiri bita $Q_3Q_2Q_1Q_0$. Stanje TAP kontrolera menja se pri rastućoj ivici takt-nog signala TCK saglasno dijagramu prikazanom na sl. 3. Naredno stanje određeno je trenutnim stanjem i vrednošću signala TMS. Na primer, ako je TAP kontroler u stanju *Capture-DR* ($Q_3Q_2Q_1Q_0=0110$) i ako je TMS=1, pri rastućoj ivici signala TCK kontroler će preći u stanje *Exit1-DR* ($Q_3Q_2Q_1Q_0=0001$), a ako je TMS=0, kontroler će preći u stanje *Shift-DR* ($Q_3Q_2Q_1Q_0=0010$).

Radom testne logike upravlja se dovođenjem odgovarajućeg niza logičkih nula i jedinica na TMS ulaz. Na primer, ako želimo da TAP kontroler dovedemo u stanje *Shift-DR* iz stanja *Test-Logic-Reset*, potrebno je u četiri taktna intervala signala TCK na TMS ulaz dovesti sledeći niz nula i jedinica "0100". Kako se testna logika okida rastućom ivicom signala TCK, poželjno je da se vrednost signala TMS menja pri opadajućoj ivici signala TCK. Isto pravilo važi i za signal TDI.

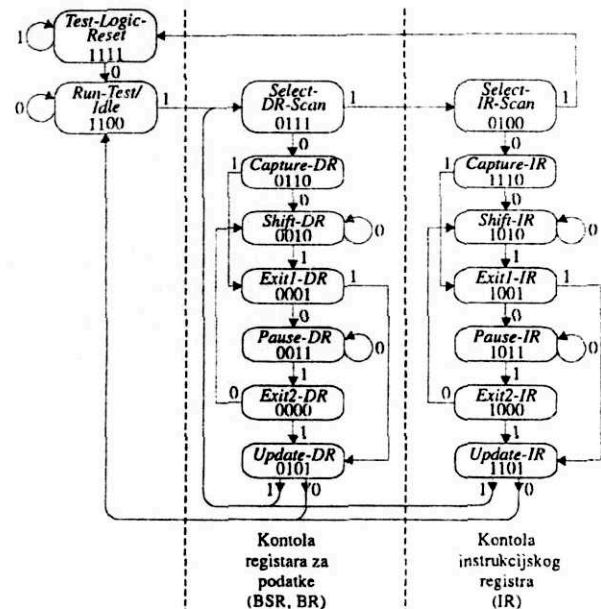
Izbor testnog režima vrši se učitavanjem odgovarajuće instrukcije u IR. Prema BS1149.1, postoje tri obavezne instrukcije: *sample/preload*, *bypass* i *extest*. Osim njih, ugrađićemo i *intest* instrukciju. Ovim instrukcijama odgovaraju četiri testna režima: *sample/preload režim*, *bypass režim*, *extest režim* i *intest režim*. Da bi se instrukcija upisala u IR, potrebno je TAP kontroler dovesti u stanje *Shift-IR* dovođenjem na ulaz TMS odgovarajućeg niza nula i jedinica. Time se između TDI ulaza i TDO izlaza postavlja IR. Zatim

se sa TDI ulaza učitava u IR kôd instrukcije prema tabl. 1.

Tabl. 1 - Kôdovi ugrađenih instrukcija

Instrukcija	Kôd
extest	00
sample/preload	10
intest	01
bypass	11

Da bi se primenio neki testni režim potrebno je: 1) dovesti TAP kontroler u poznato početno stanje *Test-Logic-Reset*, 2) učitati željenu instrukciju u IR i 3) učitavati podatke u izabrani registar za podatke (BSR ili BR) i/ili iščitavati podatke iz njega (zavisno od testnog režima).

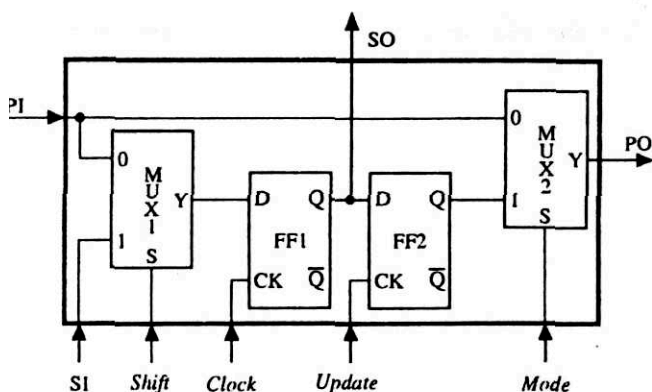


Sl. 3 - Dijagram stanja TAP kontrolera

2.1. Princip rada projektovane boundary scan ćelije

Boundary scan ćelija može se projektovati na različite načine. Način projektovanja boundary scan ćelije utiče na složenost projektovanja svih ostalih delova testne logike, tako da je boundary scan ćeliju potrebno pažljivo projektovati. Električna šema projektovane boundary scan ćelije prikazana je na slici 4. Ćelija ima serijski ulaz SI, serijski izlaz SO, paralelni ulaz PI i paralelni izlaz PO. Kontrolnim signalima *Shift*, *Clock*, *Update* i *Mode*, koji se generišu u TAP kontroleru, bira se način rada ćelije. Ćelija se sastoji od dva D flip-flopa i dva multiplexera 2 u 1. Flip-flipovi se okidaju rastućom ivicom na CK ulazu. Prenos informacije sa ulaza boundary scan ćelije na njen paralelni izlaz PO vrši se uvek u dva koraka. Informacija dovedena na PI ili SI ulaz prvo se učitava u interni flip-flop FF1, odakle je direktno dostupna na SO izlazu, ali ne i na PO izlazu.

Da bi se učitana informacija dovela na paralelni izlaz PO, u narednom koraku se stanje sa izlaza FF1 upisuje u FF2.



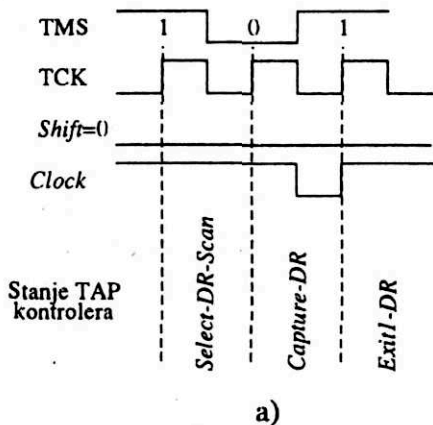
Sl. 4 - Električna šema boundary scan ćelije

Kada je *Mode*=0, PI ulaz je preko MUX2 vezan za PO izlaz i ćelija je transparentna za paralelni protok sistemskih signala. *Signal Mode* ima vrednost 1 u dva slučaja: 1) kada je u IR učitana instrukcija *extest* i 2) kada je u IR učitana instrukcija *intest*. Inače je *Mode*=0.

Signal *Shift* uzima vrednost 1 pri opadajućoj ivici signala TCK kad je TAP kontroler u stanju *Shift-DR*, zadržava tu vrednost sve dok je kontroler u stanju *Shift-DR*, a vraća se na 0 pri opadajućoj ivici signala TCK kad je kontroler u stanju *Exit1-DR*. Inače je *Shift*=0.

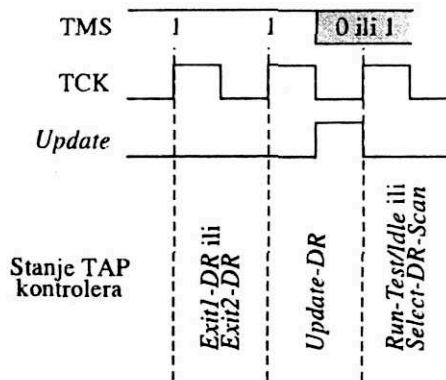
Signal *Clock* ima isti talasni oblik kao testni takt TCK ako je TAP kontroler u stanju *Capture-DR* ili u stanju *Shift-DR*. Inače je *Clock*=1. Talasni oblici signala *Shift* i *Clock* prikazani su na slikama 5a i 5b.

Shodno ovome, kad TAP kontroler prelazi iz stanja *Capture-DR* u naredno stanje (svejedno da li je to *Shift-DR* ili *Exit1-DR*), informacija sa paralelnog ulaza PI učitava se u FF1. Kad TAP kontroler prelazi iz stanja *Shift-DR* u naredno stanje (svejedno da li je to *Shift-DR* ili *Exit1-DR*), informacija sa serijskog ulaza SI učitava se u FF1.



Sl. 5 - Talasni oblici signala Shift i Clock

Signal *Update* stalno je na logičkoj nuli osim u drugoj poluperiodi taktnog intervala u toku koga je TAP kontroler u stanju *Update-DR*. U ovoj poluperiodi na signalu *Update* javlja se pozitivan impuls koji okida FF2. Može se reći da se FF2 okida opadajućom ivicom signala TCK kada je TAP kontroler u stanju *Update-DR*. Talasni oblik signala *Update-DR* prikazan je na sl. 6.

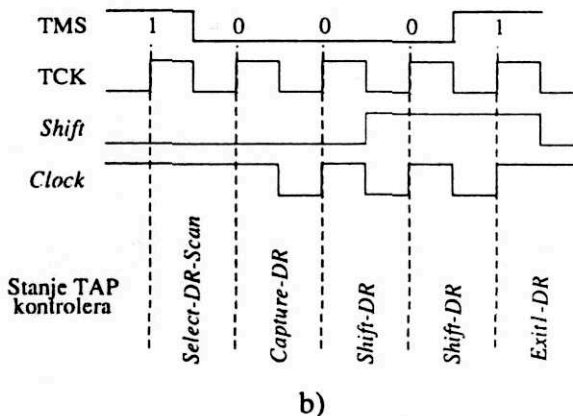


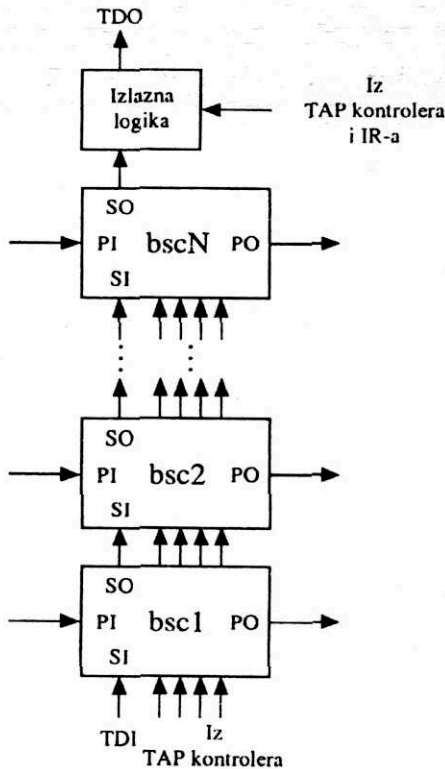
Sl. 6 - Talasni oblik signala Update

Način povezivanja N boundary scan ćelija u boundary scan registar prikazan je na sl. 7. Serijski ulaz prve boundary scan ćelije vezan je za TDI, a serijski izlaz poslednje (N-te) boundary scan ćelije je preko izlazne logike vezan za TDO.

2.2. Princip rada projektovanog bypass registra

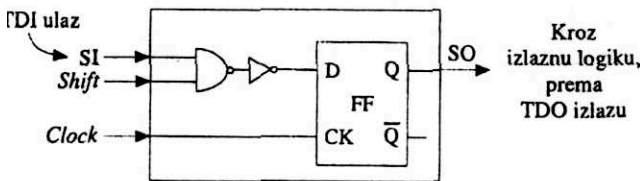
Električna šema projektovanog bypass registra prikazana je na sl. 8. BR ima serijski ulaz SI i serijski izlaz SO. Kontrolnim signalima *Shift* i *Clock* koji se generišu u TAP kontroleru upravlja se radom registra. Moguć je samo serijski upis i pomeranje informacije kroz BR. Flip-flop FF okida se rastućom ivicom na CK ulazu. Da bi bio moguć serijski upis sa SI ulaza, potrebno je da su istovremeno ispunjena dva uslova: 1) da signal *Shift* bude na logičkoj jedinici i 2) da se javi rastuća ivica na signalu *Clock*. Ova dva uslova mogu da budu istovremeno ispunjena samo kada TAP kontro-





Sl. 7 - Povezivanje N boundary scan ćelija u boundary scan registar

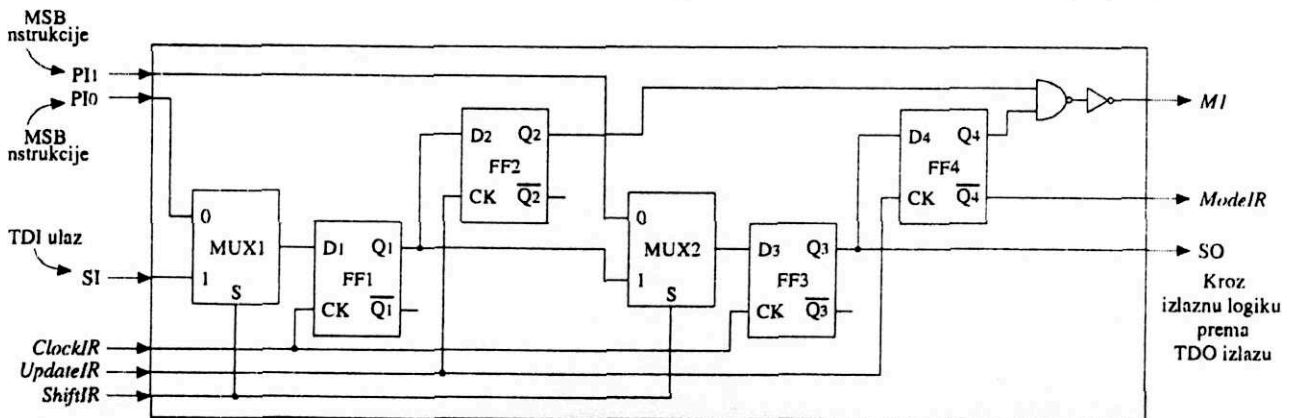
Iz prelazi iz stanja Shift-DR u stanje Exit1-DR ili u stanje Exit1-DR, kao što je prikazano na sl. 5b.



Sl. 8 - Električna šema bypass registra

2.3. Princip rada projektovanog instrukcijskog registra

U instrukcijski registar upisuje se instrukcija kojom se određuje način rada testne logike. Električna



Sl. 9 - Električna šema instrukcijskog registra

šema projektovanog instrukcijskog registra prikazana je na sl. 9. IR ima serijski ulaz SI, paralelne ulaze PI0 i PI1, i serijski izlaz SO. Dakle, moguć je serijski i paralelni upis instrukcije u IR. Serijski ulaz IR-a vezan je za TDI ulaz, a paralelni ulazi vezani su za dva izabrana primarna ulaza čipa. Radom IR-a upravljaju kontrolni signali ShiftIR, ClockIR i UpdateIR koji se generišu u TAP kontroleru. Kao i boundary scan registar, IR je dvostruki registar i upis informacije u njega odvija se u dva koraka. U prvom koraku flip-flopovi FF1 i FF3 prihvataju informaciju sa serijskog ili paralelnog ulaza, a zatim se, u drugom koraku, informacija upisuje u izlazne flip-flopove FF2 i FF4. Flip-flopovi se okidaju rastućom ivicom na CK ulazu.

Izlaz IR-a su i dva kontrolna signala: MI i ModeIR. Signal MI je na logičkoj jedinici jedino ako je u IR upisana bypass instrukcija "11". Inače je MI=0. Signal ModeIR je na logičkoj jedinici ako je u IR upisana instrukcija intest ili instrukcija extest, dakle kada je testna logika aktivna, a sistemska logika izolovana od primarnih ulaza i primarnih izlaza. Ako je u IR upisana instrukcija bypass ili instrukcija sample/preload, signal ModeIR je na logičkoj nuli.

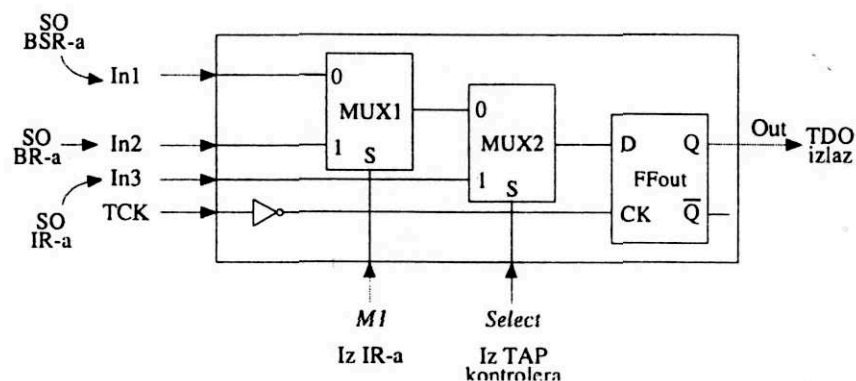
Signali ShiftIR, ClockIR i UpdateIR generišu se slično signalima Shift, Clock i Update za boundary scan ćeliju. Talasni oblici ovih signala dobijaju se tako što se na sl. 5a, 5b i 6 umesto stanja TAP kontrolera kada je ovaj u radu sa registrima za podatke zamene odgovarajućim stanjima za rad sa IR-om. Na primer, talasni oblici signala ShiftIR i ClockIR odgovaraju onim prikazanim na sl. 5a ako se na sl. 5a uvedu sledeće izmene: umesto Shift treba da stoji ShiftIR, umesto Clock treba da stoji ClockIR, umesto Select-DR-Scan treba da stoji Select-IR-Scan, umesto Capture-DR treba da stoji Capture-IR, a umesto Exit1-DR treba da stoji Exit1-IR.

2.4. Princip rada projektovane izlazne logike

Električna šema izlazne logike prikazana je na sl. 10. Izlazna logika prihvata serijske izlaze iz BSR-a (SO boundary scan ćelije bsc4), BR-a i IR-a i jedan od njih vodi na TDO izlaz preko izlaznog flip-flopa FFout. Flip-flop FFout okida se opadajućom ivicom TCK signala kako bi se skratilo vreme kašnjenja signala kroz

izlaznu logiku. Svi ulazni signali postaju validni pri rastućoj ivici TCK signala, tako da D ulaz flip-flopa postaje validan posle kašnjenja dva multipleksera. Frekvencija testnog takta TCK bira se tako da poluperioda signala TCK bude duža od sume kašnjenja dva multipleksera i vremena postavljanja flip-flopa. Tada je moguće flip-flop okidati opadajućom ivicom signala TCK tako da ukupno kašnjenje izlazne logike bude jednako samo jednoj periodu testnog takta TCK.

Izbor aktivnog ulaza vrši se pomoću dva kontrolna signala: *M1* i *Select*. Signal *M1* generiše se u IR-u i ima vrednost $M1=1$ samo ako je u IR upisana instrukcija $bypass="11"$. Inače je $M1=0$. Signal *Select* generiše se u TAP kontroleru i ima vrednost $Select=Q3$. To znači da će u svim stanjima TAP kontrolera kod kojih je najviši bit $Q3$ na logičkoj jedinici serijski izlaz IR-a biti vezan na izlazni flip-flop, to jest na TDO izlaz. Analizom dijagrama stanja TAP kontrolera (sl. 3) očigledno je da se ovo dešava u dva slučaja: 1) kada je testna logika neaktivna (stanja *Test-Logic-Reset* i *Run-Test/Idle*) i 2) kada se radi sa IR-om (stanja *Capture-IR*, *Shift-IR*, *Exit1-IR*, *Pause-IR*, *Exit2-IR* i *Update-IR*). Pri radu sa registrima za podatke, signal *Select* je u stanju logičke nule, što dozvoljava prenos signala sa izlaza jednog od registara za podatke prema TDO izlazu.



Sl. 10 - Električna šema bypass registra

2.5. Princip rada projektovanog TAP kontrolera

Ulazi TAP kontrolera su TMS i TCK, a izlazi su sledeći kontrolni signali za BSR, BR, IR i izlaznu logiku: *ModeTAP*, *Select*, *ShiftIR*, *ClockIR*, *UpdateIR*, *Shift*, *Clock* i *Update*. Signal *ModeTAP* ima vrednost 0 jedino ako je TAP kontroler u stanju *Test-Logic-Reset* ($Q_3Q_2Q_1Q_0=1111$). Princip generisanja i talasni oblici ostalih kontrolnih signala već su ranije objašnjeni.

Dijagram stanja TAP kontrolera prikazan na sl. 3 sastoji se od tri celine. Prvu čine stanja *Test-Logic-Reset* i *Run-Test/Idle*. U ovim stanjima TAP kontroler može da ostane proizvoljno dugo. Kada je TAP kontroler u stanju *Test-Logic-Reset*, testna logika je transparentna za systemske signale, odnosno čip se ponaša kao da testna logika ne postoji. Sa sl. 3 lako je uočljivo da je potrebno maksimalno 5 rastućih ivica testnog takta u toku kojih je na ulazu TMS prisutna logička

jedinica da bi se TAP kontroler iz proizvoljnog stanja vratio u stanje *Test-Logic-Reset*. Ova upravljačka sekvenca uvek se dovodi na ulaze kola pri uključanju napajanja kako bi se TAP kontroler doveo u poznato početno stanje. Kad jednom dospe u stanje *Test-Logic-Reset*, TAP kontroler ostaje u ovom stanju sve dok je $TMS=1$ ili sve dok je signal TCK neaktivan.

Stanje *Run-Test/Idle* koristi se u toku testiranja i to kada je potrebno: 1) da se sačeka na odziv logike koja se testira i 2) kada je potrebno da testna logika bude u neaktivnom stanju dok se ne odluči o daljoj akciji ili dok se postavljaju kontrolni signali na ulazima. TAP kontroler ostaje u ovom stanju sve dok je $TMS=0$ ili sve dok je signal TCK neaktivan.

Posebna grupa stanja TAP kontrolera služi za upravljanje radom instrukcijskog registra. Korišćenjem ovih stanja u instrukcijski registar se sa TDI ulaza upisuje željena instrukcija. Instrukcija se u IR može učitati serijski ili paralelno. Serijsko učitavanje instrukcije u IR vrši se u nekoliko koraka. Prvo se na TMS ulaz dovede sekvenca "01100". Ovim se TAP kontroler prevodi u stanje *Shift-IR*. Sada se na TDI ulaz dovodi kôd instrukcije (na primer "00" za *extest* instrukciju). Istovremeno sa dovodenjem poslednjeg bita kôda instrukcije, TAP kontroler se prevodi u sta-

nje *Exit1-IR* (postavljanjem $TMS=1$), a zatim u stanje *Update-IR* (ponovo $TMS=1$). U slučaju greške pri učitavanju instrukcije, iz stanja *Exit1-IR* može se prolaskom kroz stanja *Pause-IR* i *Exit2-IR* ponovo dospeti u stanje *Shift-IR* i ponoviti učitavanje instrukcije. Paralelno učitavanje instrukcije vrši se tako što se željena instrukcija dovede na primarne ulaze (u slučaju da IR ima dužinu 2, izaberu se dva primarna ulaza koji će biti korišćeni za paralelni upis u IR). Zatim se kontroler dovede u stanje *Capture-IR* (sekvenca "0110" na TMS ulazu), pa u *Exit1-IR*, te u *Update-IR* (sekvenca "11" na TMS ulazu).

Pri prolasku kroz stanje *Update-IR*, učitana instrukcija postaje validna. Iz stanja *Update-IR* može se preći u stanje čekanja (stanje *Run-Test/Idle*) ili odmah nastaviti rad sa registrima za podatke ili ponovo sa instrukcijskim registrom.

Treća grupa stanja TAP kontrolera služi za upravljanje radom registara za podatke (BSR i BR). Zavisno od instrukcije prethodno učitane u IR, moguće je serijsko ili paralelno upisivanje podataka u BSR i serijsko upisivanje podataka u BR. Da bi se ostvario serijski upis u BSR ili BR potrebno je TAP kontroler dovesti u stanje *Shift-DR*. Zatim se sa TDI ulaza učitava bit po bit, pri čemu TAP kontroler ostaje u stanju *Shift-DR* (TMS se drži na logičkoj nuli). Istovremeno sa učitavanjem poslednjeg bita podataka TAP kontroler se mora prevesti u stanje *Exit1-DR* (postavljanjem TMS=1). Ako se zatim TAP kontroler prevede u stanje *Update-DR* (TMS=1), učitani podaci postaju validni na paralelnim izlazima BSR-a. U slučaju greške pri učitavanju podataka, iz stanja *Exit1-DR* može se prolaskom kroz stanja *Pause-DR* i *Exit2-DR* ponovo vratiti u stanje *Shift-DR* i ponoviti učitavanje podataka.

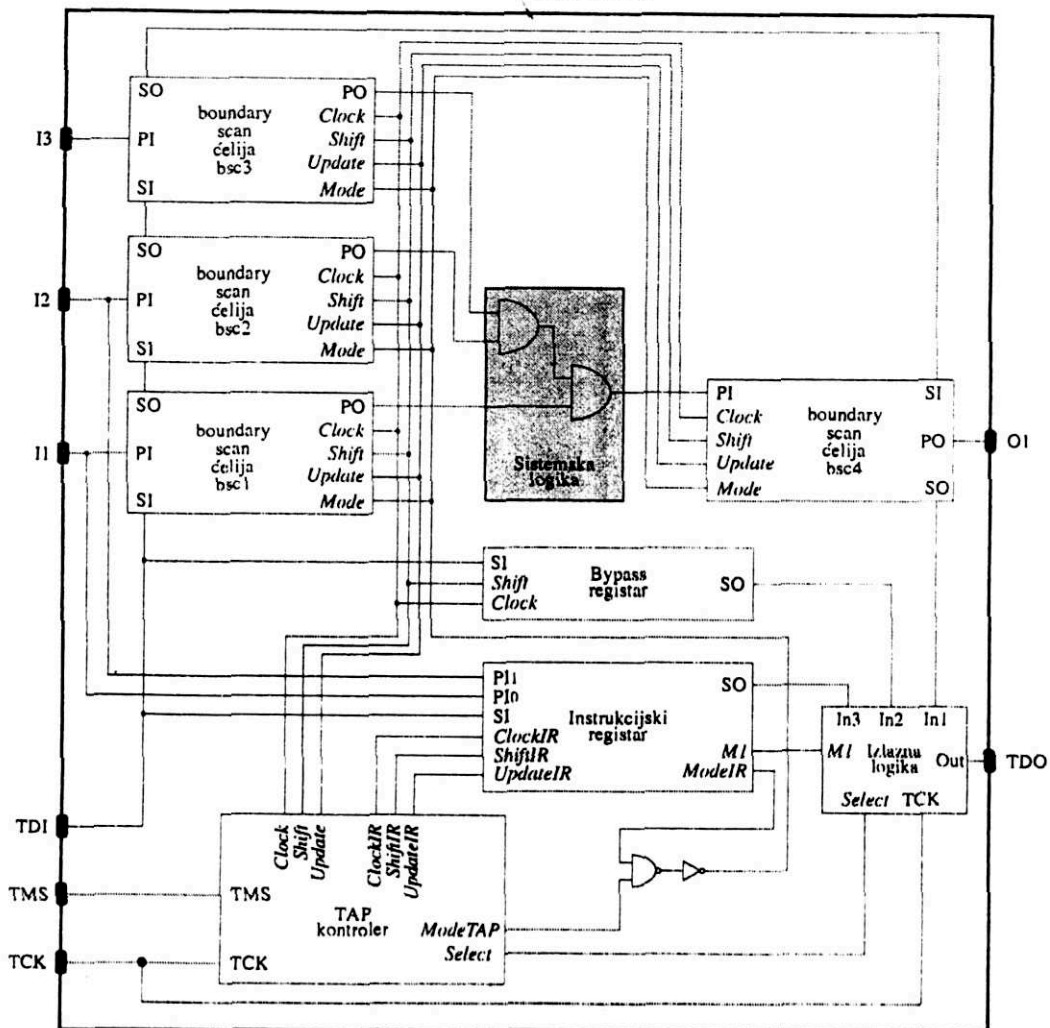
Paralelni upis u BSR vrši se tako što se TAP kontroler dovede u stanje *Capture-DR*. Zatim je moguće startovati serijsko pomeranje učitanih podataka prema TDO izlazu (prelaskom u stanje *Shift-DR*, TMS=0) ili učitane podatke prebaciti na izlaz BSR-a

prevođenjem TAP kontrolera u stanje *Exit1-DR*, pa u stanje *Update-DR* (sekvenca "11" na TMS ulazu).

2.6. Tok podataka u toku različitih testnih režima

Kada je u IR učitana instrukcija *sample/preload*, između TDI i TDO vezan je BSR. U ovom režimu rada testna logika je transparentna za signale systemske logike, ali može da ih očitava (uzorkuje) i tako omogućiti proveru ispravnog rada systemske logike. Stanja primarnih ulaza i stanja na izlazima systemske logike mogu se paralelno učitati u BSR (faza *sample*), a zatim se serijskim pomeranjem iščitati na TDO izlazu (faza *preload*).

Kada je u IR učitana *bypass* instrukcija, između TDI i TDO vezan je BR. U ovom režimu rada testna logika je transparentna za signale systemske logike, odnosno, systemska logika obavlja svoju funkciju nezavisno od testne logike. Bypass instrukcija omogućava da se čip realizovan u skladu sa BS1149.1 premesti tako da kroz njega teku testni podaci od TDI prema TDO sa kašnjenjem od samo jedne periode testnog takta TCK.



Sl. 11 - Električna šema kompletnog sistema realizovanog na štampanoj ploči

Kada je u IR učitana *extest* instrukcija, između TDI i TDO postavljen je BSR. Sistemska logika izolovana je BSR-om od primarnih ulaza i izlaza. U ovom režimu rada mogu se obaviti dve aktivnosti: 1) postavljanje željenih testnih signala na primarne izlaze čipa i 2) očitavanje stanja primarnih ulaza čipa. Pomoću ove dve aktivnosti mogu se testirati veze i/ili komponente koje nisu realizovane u skladu sa BS1149.1, ali su vezane između dve komponente sa boundary scan-om ili u povratnoj sprezi jedne komponente sa boundary scan-om.

Postavljanje stanja primarnih izlaza vrši se serijskim učitavanjem testnog vektora sa TDI ulaza u BSR. Očitavanje stanja primarnih ulaza vrši se paralelnim učitavanjem u BSR. Zatim se učitana stanja mogu serijskim pomeranjem dovesti do TDO izlaza i očitati.

Kada je u IR učitana *intest* instrukcija, između TDI i TDO postavljen je BSR. Sistemska logika izolovana je od primarnih ulaza i primarnih izlaza i njen rad je u potpunosti kontrolisan stanjima u BSR-u. Ovaj režim rada služi za testiranje sistemske logike. Testiranje se sastoji od dve aktivnosti: 1) postavljanje željenih testnih signala na ulaze sistemske logike i 2) očitavanje stanja izlaza sistemske logike.

Postavljanje stanja ulaza sistemske logike vrši se serijskim učitavanjem testnog vektora u BSR sa TDI ulaza. Očitavanje odziva sistemske logike vrši se paralelnim upisom u BSR, a zatim se učitana stanja serijskim pomeranjem dovode do TDO izlaza i iščitavaju.

3. EKSPERIMENTALNI REZULTATI

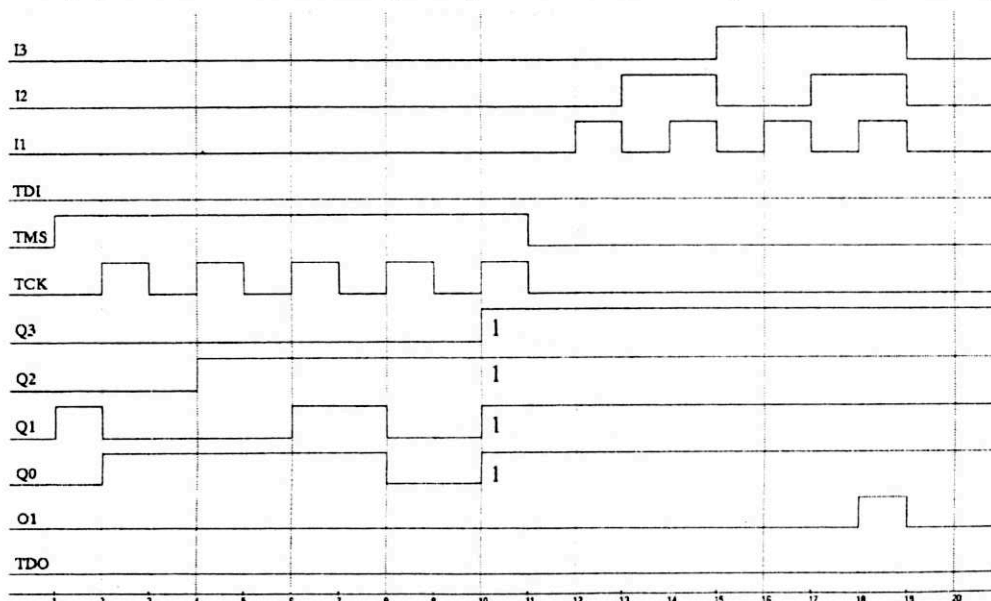
Funkcionalnost projektovane testne logike verifikovana je logičkom simulacijom [7, 8]. Zatim je na štampanoj ploči, korišćenjem diskretnih CMOS komponenti, realizovana električna šema prikazana je na sl. 11. Izabrana je jednostavna sistemska logika koja

obavlja funkciju trouzalnog I kola. Sistem ima tri primarna ulaza: I1, I2 i I3 i jedan primarni izlaz O1. Paralelni ulazi instrukcijskog registra vezani su za primarne ulaze I1 i I2. Signal *Mode* za boundary scan ćelije formiran je "I" logičkom operacijom nad signalima *ModeIR* i *ModeTAP*. Signal *Mode* je na logičkoj nuli u tri slučaja: 1) kada je TAP kontroler u stanju *Test-Logic-Reset*, (tada je *ModeTAP*=0), 2) kada je u IR učitana instrukcija *bypass* (tada je *ModeIR*=0) i 3) kada je u IR učitana instrukcija *sample/preload* (tada je *ModeIR*=0). U svim ovim slučajevima boundary scan ćelije su transparentne za sistemske signale.

U nastavku će biti izloženi rezultati analize rada projektovanog sistema u normalnom radnom režimu i pri testiranju defekata koji se manifestuju pojavom permanentnog stanja u nekom čvoru unutar sistemske logike. Na ulaze kola dovodeni su upravljački signali iz računara, a izlazi kola su vođeni u računar radi grafičkog prikaza i analize. Za komunikaciju između računara i štampane ploče korišćen je 96-bitni paralelni digitalni ulazno/izlazni interfejs PC-DIO-96/PnP. Upravljački softver za ovaj interfejs razvijen je u programskom jeziku C korišćenjem NI-DAQ biblioteke drajvera [9].

3.1. Normalan rad sistema

Pri uključenju napajanja TAP kontroler može da se nađe u proizvoljnom početnom stanju. Stoga se prvo izvršava komandna sekvenca kojom se kontroler dovodi u stanje *Test-Logic-Reset*. Dakle, u toku najmanje 5 perioda testnog takta TCK signal TMS se drži u stanju logičke jedinice, a zatim se postavi na logičku nulu. Bez obzira na trenutni sadržaj instrukcijskog registra, kada je TAP kontroler u stanju *Test-Logic-Reset* sistemska logika funkcioniše nezavisno od testne logike. Dovođenjem na ulaze I1, I2 i I3 svih mogućih pobudnih vektora može se proveriti funkcionisanje sistemske



Sl. 12 - Dovodjenje TAP kontrolera u stanje *Test-Logic-Reset* i testiranje funkcije sistemske logike

logike. Na sl. 12 prikazani su talasni oblici ulaznih i izlaznih signala, kao i promenljivih stanja TAP kontrolera Q_3, Q_2, Q_1 i Q_0 . Sa talasnih oblika se vidi da je TAP kontroler doveden u stanje *Test-Logic-Reset* ($Q_3Q_2Q_1Q_0=1111$), a zatim su na ulaze I1, I2 i I3 dovedene sve moguće kombinacije stanja 0 i 1. Lako se može uočiti da sistemka logika obavlja funkciju trouzalnog I kola.

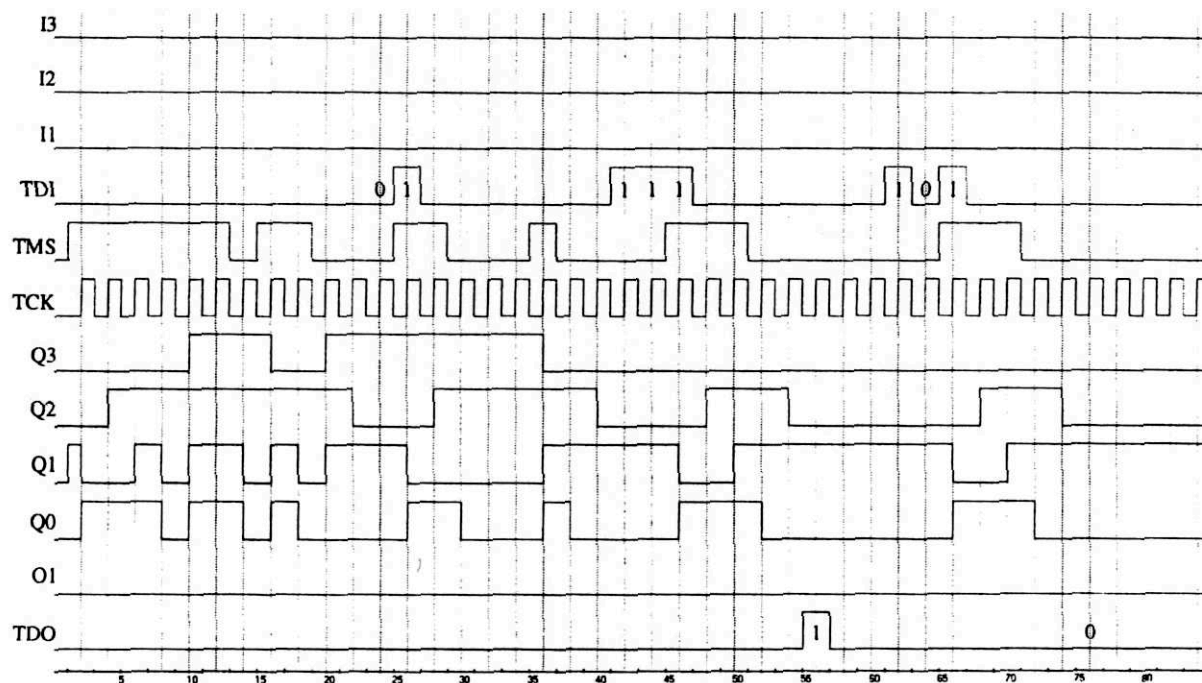
3.2. Intest testni režim

Intest testni režim služi za testiranje sistemke logike. Na sl. 13 dati su snimljeni talasni oblici signala u ovom režimu rada. TAP kontroler je doveden u stanje, odakle *Test-Logic-Reset* odakle je preveden u stanje *Shift-IR*, učitana je *intest* instrukcija dovođenjem sekvence "01" na TDI ulaz, a zatim je TAP kontroler preveden u stanje *Run-Test/Idle*. U tom stanju si-

stemska logika i primarni izlaz O1 pobuđuju se iz BSR-a.

Da bi se na ulaze sistemke logike doveo testni vektor, TAP kontroler je preveden u stanje *Shift-DR*, sa TDI ulaza u tri periode signala TCK učitana je testni vektor $t_7="111"$. Pri učitavanju poslednje jedinice, TAP kontroler je preveden u stanje *Exit1-DR*, a zatim u stanje *Update-DR*. Pri opadajućoj ivici TCK signala, dok je TAP kontroler u stanju *Update-DR*, učitani testni vektor pojavljuje se na izlazima boundary scan ćelija bsc1, bsc2 i bsc3, odnosno na ulazima sistemke logike. Zatim, prelaskom u stanje *Run-Test/Idle*, TAP kontroler fiksira ovakvo stanje u kolu.

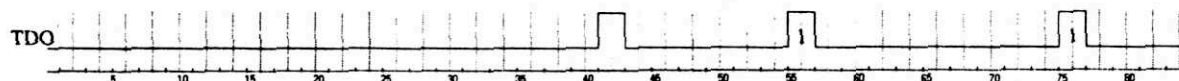
Ako je sistemka logika bez defekata, njen izlaz (ulaz ćelije bsc4) biće u stanju 1. Da bi se stanje izlaza sistemke logike očitalo, TAP kontroler se prevodi u stanje *Capture-DR*, pa u stanje *Shift-DR*. U trenutku ulaska u stanje *Shift-DR* stanja primarnih ulaza I1, I2 i



a)



b)



c)

Sl. 13 - a) Talasni oblici signala u intest testnom režimu kada u kolu nema defekata; b) TDO signal u prisustvu permanentne nule na izlazu sistemke logike; c) TDO signal u prisustvu permanentne jedinice na izlazu sistemke logike

I3 ("000" u tom trenutku) učitavaju se u boundary scan ćelije bsc1, bsc2 i bsc3, a stanje izlaza sistemske logike ('1' u tom trenutku) učitava se u bsc4. Nadalje TAP kontroler ostaje u stanju *Shift-DR* nekoliko perioda TCK signala (minimalno jedna perioda TCK) kako bi se učitano stanje iz bsc4 dovelo do TDO izlaza. Sa sl. 13a se ponovo potvrđuje da je kašnjenje od ulaska u stanje *Shift-DR*, pa do pojave logičke jedinice na TDO izlazu (kašnjenje izlazne logike) $T_{read}(O1) = T_{TCK}/2$.

Zatim je TAP kontroler još jednom proveden kroz isti testni ciklus (dovođenje testnog vektora i očitavanje odziva), pri čemu je na ulaze sistemske logike primenjen testni vektor $t_5 = "101"$, a na TDO izlazu očitana logička nula.

Ako se na izlazu sistemske logike (ulaz boundary scan ćelije bsc4) izazove *defekt permanentne logičke nule*, na TDO izlazu dobija se talasni oblik prikazan na sl. 13b. Vidi se da je kao odziv na testni vektor $t_7 = "111"$ na TDO izlazu očitana logička nula umesto logičke jedinice, čime je defekt detektovan.

Ako se na izlazu sistemske logike (ulaz ćelije bsc4) izazove *defekt permanentne logičke jedinice*, na TDO izlazu dobija se talasni oblik prikazan na sl. 13c. Vidi se da je kao odziv na testni vektor $t_5 = "101"$ na TDO izlazu očitana logička jedinica umesto logičke nule, čime je defekt detektovan.

4. ZAKLJUČAK

Projektovanje osnovnih gradivnih elemenata testne logike za implementaciju BS1149.1 standarda omogućava realizaciju proizvoljnog integrisanog kola ili diskretnog sistema baziranog na ovom standardu. Eksperimenti izvršeni na diskretnom sistemu realizovanom korišćenjem razvijene testne logike dokazuju njenu funkcionalnost u normalnom radu i u četiri testna režima: *sample/preload*, *bypass*, *extest* i *intest*. Naša dalja istraživanja usmerena su na projektovanje i proizvodnju integrisanih kola korišćenjem razvijene

testne logike. Takođe, u cilju sprežanja više čipova realizovanih prema BS1149.1 u složeniji sistem, potrebno je razviti dodatnu logiku za upravljanje testiranjem pojedinih čipova. To podrazumeva i ugradnju dodatnih instrukcija za identifikaciju čipova, kao i eventualnu ugradnju mehanizama za samotestiranje (BIST - built-in self test) u čipove.

LITERATURA

- [1] R. G. Bennetts, **Introduction to Digital Board Testing**, Computer Systems Engineering Series, Edward Arnold (Publishers) Ltd., London, UK, 1982.
- [2] B. R. Wilkins, **Testing Digital Circuits**, Van Nostrand Reinhold, Wokingham, UK, 1986.
- [3] M. Abramovici, M. Breuer, A. Friedman, **Digital System Testing and Testable Design**, Computer Science Press, New York, USA, 1990.
- [4] H. Bleeker, P. v. d. Eijnden, F. D. Jong, **Boundary-Scan Test - A Practical Approach**, Kluwer Academic Publishers, Dordrecht, Germany, 1993.
- [5] IEEE Std 1149.1 (including IEEE Std 1149.1a), **IEEE Standard Test Access Port and Boundary-Scan Architecture**, IEEE Computer Society, New York, USA, 1993.
- [6] **Supplement to IEEE Std 1149.1-1990, IEEE Standard Test Access Port and Boundary-Scan Architecture**, IEEE, New York, USA, 1995.
- [7] V. M. Panić, S. Janković, D. Milovanović, V. B. Litovski, "Cell Design for Boundary-Scan Implementation", Proc. of the 22th Int. Conf. on Microelectronics, Niš, September 1999.
- [8] V. M. Panić, **Razvoj integrisanih kola zasnovan na primeni 'boundary scan' koncepta**, magistarska teza, Univerzitet u Nišu, Elektronski fakultet, Niš, oktobar 1999.
- [9] **NI-DAQ for PC Compatibles Version 6.1, Release Notes**, National Instruments, Part Number 321647C-01, April 1998.

SUMMARY

IMPLEMENTATION OF DIGITAL ELECTRONIC CIRCUITS COMPLYING TO THE BOUNDARY SCAN STANDARD

Boundary scan concept of design for testability assumes the incorporation of additional test logic into the chip or printed circuit board that simplifies the testing and diagnostics of the defects in the system after production, as well as during exploitation and maintenance. This paper presents a practical implementation of boundary scan standard for digital systems testing. Necessary test logic is designed, verified by simulation, implemented on a printed board using discrete CMOS components and its efficiency in testing of stuck-at defects is proved.