علوم و مهندسي كامپيوتر
نشريه علمي پژوهشي انجمن كامپيوتر ايران
مجلد ۱، شماره ۳ (الف)، پائيز۱۳۸۲
صفحات ۳۳-۴۱
مقاله كوتاه

ارائه يك روش جديد تخصيص ثبات براي بهبود آزمون پذيري در سنتز رفتاري

سعيد صفري* اميرحسين جهانگير* هادي اسمعيل‌زاده**

* دانشكده مهندسي كامپيوتر، دانشگاه صنعتي شريف، تهران ، ايران
** گروه مهندسي برق و كامپيوتر، دانشكده فني, دانشگاه تهران، تهران, ايران

چكيده

در نظر گرفتن ملاحظات آزمون‌پذيري در طي مراحل سنتز رفتاري، موجب كاهش سربار آزمون‌پذيري، كاهش هزينه‌هاي آزمون و نيز افزايش كيفيت آزمون‌پذيري مدار نهايي مي‌شود. در اين مقاله يک الگوريتم تخصيص ثبات جديد مبتني بر رنگ‌آميزي گراف وزن‌دار با هدف بهبود آزمون‌پذيري مدار ارائه مي‌شود. در اين روش تخصيص ثبات، معيارهاي آزمون‌پذيري سطح بالا مانند عمق ترتيبي، حلقه‌ي ترتيبي، قابليت كنترل و قابليت مشاهده در نظر گرفته مي‌شود. نتايج حاصل از آزمايش‌ها نشان مي‌دهد كه روش تخصيص ثبات ارائه شده سبب كاهش قابل ملاحظه‌ي زمان توليد بردار آزمون و نيز افزايش پوشش خطا مي‌شود

كلمات كليدي: سنتز رفتاري، تخصيص ثبات، گراف تنازع، رنگ‌آميزي گراف وزن‌دار

۱- مقدمه

با افزايش روزافزون پيچيدگي مدارهاي مجتمع و نيز افزايش تقاضا براي كاهش زمان تحويل به بازار، روند طراحي به سمت استفاده از توصيف‌هاي سطح بالا سوق پيدا كرده است. در اين سطح از توصيف، طراح بدون اين‌كه مشغول جزئيات پياده‌سازي شود، تنها به توصيف رفتار سيستم مي‌پردازد.

اين سطح از طراحي بدون كمك روش‌ها و ابزارهاي طراحي كامپيوتري[۱] ممكن نيست. استفاده از اين ابزارها علاوه بر كاهش چشمگير زمان طراحي، منجر به بهبود كيفيت سخت‌افزار نهايي نيز مي‌شود.

ابزار سنتز رفتاري[۲] برنامه‌ي كامپيوتري است كه يک توصيف سطح بالا را به يک مدار در سطح انتقال ثبات[۳] تبديل مي‌كند و در همين حال برخي از معيارهاي طراحي مانند مساحت، تاخير، آزمون‌پذيري و توان مصرفي را بهينه مي‌كند [۱, ۲].

ابزارهاي سنتز رفتاري سنتي فقط معيارهاي مساحت و تأخير را بهينه مي‌كنند، از اين‌رو مدار توليد شده توسط اين ابزارها لزوماً از نظر آزمون‌پذيري بهينه نيستند. بنابراين براي افزايش آزمون‌پذيري مدار نهايي، بايد پس از خاتمه‌ي سنتز رفتاري روش‌هاي افزايش آزمون‌پذيري را به مدار RTL توليد شده اعمال نمود [۳]، كه اين امر مي‌تواند منجر به افزايش قابل ملاحظه‌ي سربار آزمون شده و كيفيت مدار نهايي را به نحو چشمگيري كاهش دهد.

براي حل اين مشكل بايد آزمون‌پذيري را به‌صورت يكي از معيارهاي طراحي سطح بالا در نظر گرفت.

هدف از سنتز رفتاري آزمون‌پذير[۴]، سنتز يک معماري آزمون‌پذير است كه در سطوح پايين‌تر به هيچ‌گونه سرباري براي افزايش آزمون‌پذيري نياز نداشته باشد. به دليل عدم وجود مدل خرابي مناسب در اين سطح كه با عيب‌هاي ساخت مرتبط باشد، عمليات سنتز رفتاري آزمون‌پذير پيچيده مي‌باشد. به همين دليل روش‌هاي نويني به منظور دخالت دادن معيارهايي مانند ATPG[۵] و BIST[۶] در طي مراحل سنتز رفتاري ارائه شده‌اند. معيارهاي آزمون‌پذيري مختلفي مانند عمق ترتيبي[۷]، حلقه‌ي ترتيبي[۸]، قابليت كنترل/مشاهده[۹] و حلقه به خود[۱۰] بر روي هزينه‌هاي ATPG و BIST تاثير مي‌گذارد.

در مراجع [۴-۷] مروري بر سنتزرفتاري آزمون‌پذير ارائه شده است. مهمترين معيارهاي آزمون‌پذيري كه در بيشتر ابزارهاي سنتز رفتاري مورد توجه قرار مي‌گيرند عمدتا شامل عمق ترتيبي، حلقه‌ي ترتيبي و قابليت كنترل/ مشاهده

مـي‌باشـد [۷]. در مراجـع [۹,۸]، کـاهش حلقـه بـه خـود در مسـير داده بـا هـدف حـل مشـكلات BIST ارائـه شـده اسـت. در [۸] دو روش تخصـيص ارائـه شـده اسـت كـه يـك گـراف جريـان داده‌ي زمانبنـدي شـده را بـه يـك مسيرداده‌ي آزمـون‌پـذير نگاشـت مـي‌كنـد. روش اول مبتنـي بـر الگـوريتم‌هـاي مكاشـفه‌اي گـراف و روش دوم مبتنـي بـر برنامـه‌ريـزي خطـي صـحيح اسـت. در [۱۱,۱۰] روش‌هـايي بـراي بـه اشـتراك گذاشـتن سـخت‌افـزار ارائـه شـده اسـت كه هدف آن توليـد مـداري اسـت كـه سـربار لازم بـراي زنجيـره‌ي پـويش جزئـي آن كمينـه باشـد. ايـن روش حلقـه‌هـاي مسـتقل را كـه در گـراف جريـان داده-كنترل تشكيل شده اسـت پيـدا مـي‌كنـد و سـپس بـا كمتـرين تعـداد ثبـات‌هـاي پـويش ايـن حلقـه‌هـا را مـي‌شـكند. يـك روش سـنتز رفتـاري آزمـون‌پـذير در مرجـع [۱۲] ارائـه شـده اسـت. ايـن روش بـا در نظـر گـرفتن آزمـون‌پـذيري در مرحلـه‌ي تخصـيص منبـع و ثبـات مـداري توليـد مـي‌كنـد كـه آزمـون آن بـه سـادگي انجـام مـي‌شـود. در مراجـع [۱۵,۱۴,۱۳] بهبـود قابليـت كنتـرل/مشـاهده در طـي مراحـل سـنتز رفتـاري در نظـر گرفتـه مـي‌شـود. در [۱۸,۱۷,۱۶] روشـي بـراي آزمـون نقـاط سخت‌آزمـون[۱۱] ارائـه شـده اسـت كـه مبتني بر افزودن نقـاط آزمـون اسـت. تمـام ايـن روش‌هـا بـر ايـن اسـاس اسـتوار هسـتند كـه مرحلـه‌ي زمانبنـدي انجـام شـده اسـت و ايـن الگـوريتم‌هـا بـر روي گـراف جريـان داده‌ي زمانبنـدي شـده اعمـال مـي‌شـوند. در مرجـع [۱۹] روشـي ارائه شده است كـه زمانبنـدي و تخصـيص منبـع را هـم‌زمـان انجـام مـي‌دهـد. در اين روش مرحلـه‌ي تخصـيص منـابع بـا در نظـر گـرفتن قابليـت كنتـرل/مشـاهده انجـام مـي‌شـود و سـپس تـلاش مـي‌شـود بـا تغييـر آن آزمـون پـذيري را بهبـود داد.

در اين مقاله يك الگوريتم جديـد تخصـيص منبـع معرفـي مـي شـود كـه مبتنـي بر رنگ‌آميـزي گـراف وزن‌دار اسـت. در حاليكـه در بيشـتر روش‌هـاي سـطح بـالا يك يا دو معيار آزمـون‌پـذيري سـطح بـالا در مراحـل سـنتز رفتـاري نظـر گرفتـه مـي‌شـوند، در روش پيشـنهادي مـا تـلاش شـده اسـت كـه بهينـه‌سـازي ايـن معيارها به‌صورت همزمان انجام شود.

سـاختار ادامـه‌ي مقالـه بـه‌صـورت زيـر اسـت. در بخـش ۲ مـروري بـر سـنتز رفتـاري و معيارهـاي آزمـون‌پـذيري ارائـه مـي‌شـود. بخـش ۳ بـه معرفـي مـدل پيشـنهادي بـراي تخصـيص ثبـات مـي‌پـردازد. در بخـش ٤ نتـايج آزمـايش‌هـاي انجام شده ارائه شده و بخش ۵ به نتيجه‌گيري اختصاص دارد.

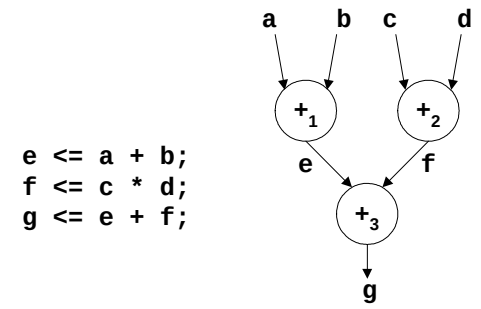
۲- مفاهيم اوليه

۲-۱ سنتز رفتاري

ابزار سـنتز رفتـاري معمـولا توصـيف سـطح بـالاي مـدار را بـه يـك قالـب ميـاني مناسـب ماننـد *گـراف جريـان داده*[۱۲] تبـديل مـي‌كنـد [۲۰]. در يـك DFG هـر گـره بـا يـك *عمليـات* و هـر لبـه بـا يـك متغيـر متنـاظر اسـت. شـكل ۱ ، يـك قطعـه كـد VHDL و DFG متنـاظر آن را نشـان مـي‌دهـد. ايـن مـدار را بـراي مراجعات بعدي *ex1* نام‌گـذاري مـي‌كنـيم. ابـزار سـنتز رفتـاري بـراي توليـد يـك معمـاري در سـطح RTL ، دو *عمليـات زمـان‌بنـدي* و *تخصـيص منبـع* را انجـام مي‌دهد.

زمانبندي: در اين مرحله با تخصيص هر *عمليات* به يك يا چند سيكل ساعت، رفتار سيكل به سيكل مدار مشخص مي‌شود. نتيجه‌ي *عمليات* زمانبندي، يك *گراف زمانبندي* شده است كه SDFG[۱۳] ناميده مي‌شود.

براي هر *عمليات* O در DFG ، زودترين (ديرترين) زماني كه *عمليات* O در آن اجرا مي‌شود را *O.Earliest* (*O.Latest*) مي‌ناميم. در يك SDFG، اولين زماني كه يك متغير تعريف مي‌شود را *زمان تولد* و آخرين زماني كه يك متغير استفاده مي‌شود را *زمان مرگ* متغير مي‌ناميم.



شكل ۱- (الف) كد VHDL مدار *ex1* , (ب) DFG متناظر آن

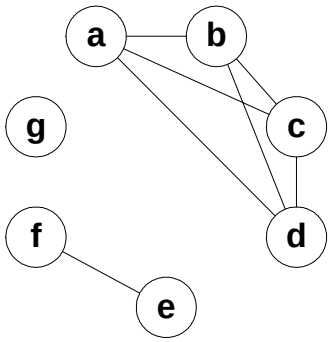
بازه‌ي زماني تولـد تـا مـرگ يـك متغيـر را *طـول عمـر* آن متغيـر مـي‌ناميم. اگـر طول *عمـر* دو متغيـر بـا هـم هـم‌پوشـاني نداشـته باشـد، آن دو متغيـر را *سـازگار* مـي‌ناميم. دو متغيـر سـازگار را مـي‌تـوان بـر روي يـك ثبـات نگاشـت كـرد. اگـر طـول *عمـر* دو متغيـر هـم‌پوشـاني داشـته باشـد، آن دو متغيـر را *ناسـازگار* مـي‌نـاميم و نمـي‌تـوان آن دو متغيـر را بـر روي يـك ثبـات نگاشـت كـرد [۷]. بـراي مثـال در مـدار *ex1* متغيرهـاي *e* و *g* سـازگار و متغيرهـاي *e* و *f* ناسازگار هستند.

تخصـيص منـابع: در مرحلـه‌ي تخصـيص منبـع كـه بـرروي يـك SDFG انجـام مـي‌شـود، پيمانـه‌هـا بـراي اجـراي *عمليـات*، ثبـات‌هـا بـراي ذخيـره‌ي متغيرهـا و منابع اتصالي بـراي ارتبـاط اجـزا تخصـيص داده مـي‌شـود. بـراي تخصـيص ثبـات الگـوريتم‌هـاي مختلفـي وجـود دارد. الگـوريتم حريصـانه LEA[۱٤] كـه در ابتـدا بـراي مسـئله‌ي مسـيريابي كانال‌هـا در سـطح تراشـه ارائـه شـد [۲۱]، مـي‌توانـد بـراي تخصـيص ثبـات بـا كمتـرين تعـداد ثبـات مـورد اسـتفاده قـرار بگيـرد. در ايـن الگـوريتم ابتـدا طـول *عمـر* متغيرهـا از روي SDFG بـه‌دسـت مـي‌آيـد و سپس بر اساس زمـان تولـد در يـك جـدول مرتـب مـي‌شـوند. الگـوريتم LEA بـا پـويش جـدول، متغيرهـاي تخصـيص نيافتـه را در اولـين سـيكل زمـاني موجـود بـه ثبـات‌هـا اختصـاص مـي‌دهـد. ايـن الگـوريتم حريصـانه كمتـرين تعـداد ثبـات‌هـاي لازم بـراي تخصـيص را پيـدا مـي‌كنـد [۲۲]. الگـوريتم LEA مي‌تواند براي تخصيص پيمانه‌ها نيز مورد استفاده قرار بگيرد.

بـراي تخصـيص ثبـات دو نـوع روش وجـود دارد كـه اولـي مبتنـي بـر افـراز خوشـه‌اي[۱۵] و ديگـري مبتنـي بـر رنـگ‌آميـزي گـراف[۱٦] اسـت. در روش اول از يك گراف بـه نـام *گـراف سـازگاري*[۱۷] اسـتفاده مـي‌شـود. در گـراف سـازگاري هـر گـره نشـان‌دهنـده‌ي يـك متغيـر اسـت. اگـر دو متغيـر بـا هـم سـازگار باشـند، گـره‌هـاي متنـاظر آنهـا در گـراف سـازگاري بـا هـم همسـايه هسـتند. در ايـن حالت مسئله‌ي تخصـيص ثبـات متنـاظر بـا انجـام افـراز خوشـه‌اي بـر روي گـراف سـازگاري اسـت. در روش دوم تخصـيص ثبـات يـك *گـراف تنـازع*[۱۸] از روي SDFG سـاخته مـي‌شـود كـه در آن هـر گـره متنـاظر يـك متغيـر اسـت. در گراف تنازع اگر دو متغير بـا هـم ناسـازگار باشـند، گـره‌هـاي متنـاظر آنهـا بـا هـم همسـايه هسـتند. در ايـن روش مسـئله‌ي تخصـيص ثبـات متنـاظر بـا رنـگ‌آميـزي گـراف تنـازع مـي‌باشـد. شـكل ۲ گـراف تنـازع مـدار *ex1* را نشـان مي‌دهد.

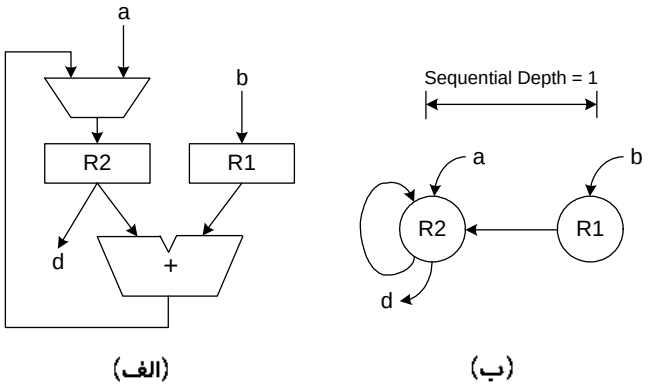
۲-۲ معيارهاي آزمون‌پذيري

قابليت كنترل/مشاهده: سادگي كنترل (مشاهده) سيگنال‌هاي ميـاني در مـدار ديجيتال را قابليت كنترل (مشاهده) مي‌ناميم. اگر يك متغير ورودي‌اصلي[۱۹] (خروجي اصلي[۲۰]) به يك ثبات تخصيص پيدا كند، اين ثبات مستقيما قابل كنترل(مشاهده) است. در غيراين‌صورت مي‌توان از طريق ساير ثبات‌ها به آن متغير دسترسي پيدا كرد.



شکل ۲- گراف تنازع مدار *ex1*

عمق ترتيبي: پس از انجام عمليات زمان‌بندي و تخصيص منابع، يك گراف به نام *گراف مداري مسير داده*[۲۱] از روي معماري سنتز شده به‌دست مي‌آيد. در DPCG هر گره نشان‌دهنده‌ي يك ثبات و هر لبه نشان‌دهنده‌ي يك مسير تركيبي بين دو ثبات است. هم‌چنين در DPCG يك لبه به هر گره‌ي متناظر با يك ثبات ورودي وارد مي‌شود و يك لبه از هر گره‌ي متناظر با يك ثبات خروجي خارج مي‌شود. شكل ۳ يك مدار RTL و DPCG متناظر آن آمده است.



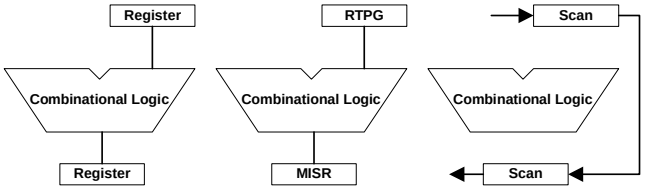
شكل ۳- (الف) يك مدار RTL , (ب) DPCG متناظر آن

مسير ترتيبي بين دو گره در DPCG نشان‌دهنده‌ي وجود يك ارتباط بين دو ثبات در مدار پياده‌سازي شده است. تعداد لبه‌ها در يك مسير ترتيبي نشان‌دهنده‌ي *عمق ترتيبي* بين دو ثبات است. از آنجا كه ممكن است چندين مسير ترتيبي بين دو ثبات وجود داشته باشد، عمق ترتيبي بين دو ثبات را كوتاه‌ترين مسير ترتيبي بين اين دو ثبات تعريف مي‌كنيم. عمق ترتيبي به‌صورت خطي بر روي هزينه‌ي ATPG تاثير مي‌گذارد [۱۳] .

از اين‌رو با كاهش عمق ترتيبي هزينه‌ي ATPG نيز كاهش مي‌يابد. براي مثال عمق ترتيبي بين ثبات‌هاي R1 و R2 در شكل ۳ برابر ۱ است.

حلقه‌ي ترتيبي: اگر در يك مسير ترتيبي گره‌هاي مبدا و مقصد يكسان باشد، طول مسير بزرگ‌تر از ۱ باشد و هيچ يك از ثبات‌هايي كه بر روي اين مسير قرار دارند ثبات ورودي/خروجي نباشند، در اين‌صورت اين مسير يك حلقه به *غيرخود*[۲۲] ناميده مي‌شود. اگر طول مسير برابر ۱ باشد، آن را حلقه به خود مي‌ناميم. طول حلقه‌ي ترتيبي به‌صورت نمايي بر روي هزينه‌ي ATPG تاثير مي‌گذارد [۷]، از اين‌رو كاهش طول حلقه‌ي ترتيبي باعث كاهش هزينهATPG می شود.

در BIST مدارهايي در سطح تراشه قرار داده مي‌شود تا بردارهاي آزمون را توليد كرده و خروجي‌هاي توليد شده را تحليل كند. براي فراهم كردن امكان توليد بردار آزمون و تحليل خروجي، ثبات‌هاي مورد استفاده در BIST بايد از نوع BILBO[۲۳] باشند. اين نوع ثبات بايد توانايي پشتيباني مدهاي كاري عادي، ورودي/خروجي پويش سريال[۲۴]، توليد بردار آزمون تصادفي[۲۵] و MISR[۲۶] را داشته باشد (شكل ۴). اگر در DPCG يك ثبات موجود باشد كه ورودي و خروجي آن مستقيما توسط يك پيمانه عملياتي به يكديگر متصل شده باشند، در معماري RTL متناظر آن يك حلقه به خود وجود دارد. چنين ثباتي را يك ثبات *خودمجاور*[۲۷] مي‌ناميم. در چنين شرايطي اگر بخواهيم از روش BIST استفاده كنيم، ثبات BILBO بايد به‌صورت هم‌زمان نقش RTPG و MISR را ايفا كند. يك ثبات با اين ويژگي را CBILBO[۲۸] مي‌ناميم. به خاطر بالا بودن هزينه‌ي مساحت و تاخير CBILBO ، بايد تا حد امكان از به‌وجود آمدن ثبات‌هاي خودمجاور در معماري اجتناب نمود.



شكل ۴- حالت‌هاي كاري مختلف ثبات BILBO

۳- مدل آزمون‌پذيري پيشنهادي

همان طوركه در بخش‌هاي قبلي توضيح داده شد، بهبود آزمون‌پذيري را مي‌توان در هر دو مرحله‌ي تخصيص پيمانه و تخصيص ثبات انجام داد. با وجود اين‌كه مدل آزمون‌پذيري پيشنهادي ما را مي‌توان در هر دو مرحله به كار برد، در اين مقاله مدل را فقط در مرحله‌ي تخصيص ثبات به‌كار مي‌بريم. در اين روش ما يك گراف وزن‌دار بنا مي‌كنيم كه گره‌هاي آن متناظر متغيرها و وزن هر لبه نشان‌دهنده‌ي هزينه‌ي نگاشت دو متغير انتهايي لبه بر روي يك ثبات است. سپس سعي مي‌كنيم رنگ‌آميزي c را به‌گونه‌اي بيابيم كه تابع $\sum_{c(u)=c(v)} w(u,v)$ را كمينه كند، در اين رابطه $w(u,v)$ نشان‌دهنده‌ي وزن لبه‌ي بين گره‌هاي u و v و $c(u)$ نشان‌دهنده‌ي رنگ گره‌ي u است.

۳-۱ گراف تنازع توسعه يافته[۲۹]

همانطوركه قبلا اشاره شد، اگر دو متغير ناسازگار باشند (يعني طول عمر آنها هم‌پوشاني داشته باشد)، نمي‌توان آنها بر روي يك ثبات نگاشت كرد. اين محدوديت را يك *محدوديت ساختاري*[۳۰] مي‌ناميم و به هيچ‌وجه نمي‌توان آن را نقض كرد (نقض محدوديت ساختاري باعث عملكرد اشتباه مدار مي شود). نقض محدوديت‌هاي آزمون‌پذيري كه به ما تحميل مي‌شود تنها باعث توليد مداري مي شود كه از ديدگاه آزمون‌پذيري كيفيت پايين‌تري دارد و باعث عمكرد اشتباه مدار نمي‌شود.

ما يك مدل مجرد به نام *گراف وزن‌دار توسعه يافته* معرفي مي‌كنيم كه در آن چهار نوع لبه وجود دارد: لبه‌هاي باوزن $\infty+$ ، لبه‌هاي با وزن مثبت، لبه های با وزن منفی و لبه های با وزن صفر.

- لبه های با وزن$\infty+$ محدوديت های ساختاری را نشان می دهند. وجود يك لبه با وزن $\infty+$ بين دو گره نشان مي‌دهد كه طول عمر متغيرهاي متناظر آنها هم‌پوشاني دارد و به همين دليل اين دو متغير نمي‌توانند بر روي يك ثبات نگاشت شوند.

- لبه‌هاي با وزن مثبت محدوديت‌هاي آزمون‌پذيري را نشان مي‌دهد. اگر بين دو گره يك لبه با وزن مثبت وجود داشته باشد، به معناي اين است كه نگاشت متغيرهاي متناظر اين دو گره بر روي يك ثبات باعث كاهش آزمون‌پذيري مدار نهايي مي‌شود. هر چه قدرمطلق وزن اين لبه بزرگتر باشد، نگاشت دو متغير انتهايي اين لبه باعث كاهش بيشتر آزمون‌پذيري مدار نهايي مي‌شود.
- اگر بين دو گره يك لبه با وزن منفي وجود داشته باشد، به معناي اين است كه نگاشت متغيرهاي متناظر اين دو گره بر روي يك ثبات باعث افرايش آزمون‌پذيري مدار نهايي مي‌شود. هر چه قدرمطلق وزن اين لبه بزرگتر باشد، نگاشت دو متغير انتهايي اين لبه باعث افرايش بيشتر آزمون‌پذيري مدار نهايي مي‌شود.
- وجود يك لبه با وزن صفر نشان‌دهنده‌ي اين است كه نگاشت دو متغير انتهايي اين لبه بر روي يك ثبات باعث افزايش يا كاهش آزمون‌پذيري مدار نهايي نمي‌شود.

يكي از ويژگي‌هاي مهم گراف ECG قابليت توسعه‌ي آن است، به اين معني كه هر نوع تمايل (يا عدم تمايل) به نگاشت دو متغير به يك رجيستر (مانند ملاحظات توان مصرفي) را مي‌توان به صورت يك لبه با وزن منفي (يا مثبت) بين گره‌هاي متناظر آنها در گراف ECG نشان داد.

در طي مراحل ساخت گراف ECG ، ممكن است چندين لبه بين دو متغير اضافه شود. بنابراين پيش از اجراي الگوريتم رنگ‌آميزي گراف وزن‌دار، بايد گراف ECG را پالايش نمود. در مرحله‌ي پالايش تمام لبه‌هاي موجود بين دو گره با يك لبه جايگزين مي‌شود. وزن لبه‌ي جديد برابر حاصل‌جمع وزن‌دار لبه‌هاي موجود بين اين دو گره است. پس از انجام عمل پالايش، مسئله‌ي تخصيص ثبات به صورت يك مسئله‌ي رنگ‌آميزي گراف وزن‌دار در مي‌آيد به گونه‌اي كه مجموع وزن‌هاي هر مجموعه‌ي رنگي كمينه باشد.

۲-۳ استخراج وزن‌هاي گراف ECG

بهبود قابليت كنترل/مشاهده: متغيرهاي موجود در SDFG را مي‌توان به سه دسته‌ي ورودي‌هاي اصلي (PI)، خروجي‌هاي اصلي (PO) و متغيرهاي مياني تقسيم نمود. اگر هر يك از متغيرهايي كه به يك ثبات تخصيص يافته PI (PO) باشد، آن ثبات مستقيما قابل كنترل (مشاهده) خواهد بود. ساير ثبات‌ها فقط از طريق اين ثبات‌ها قابل دسترسي هستند. براي افزايش قابليت كنترل (مشاهده) مدار نهايي، تلاش مي‌كنيم به هر ثبات حداقل يك PI (PO) را تخصيص دهيم.

براي افزايش قابليت كنترل، بين هر PI و هر متغير مياني سازگار با آن يك لبه با وزن $-w_{co}$ قرار مي‌دهيم. به همين ترتيب براي افزايش قابليت مشاهده بين هر PO و هر متغير مياني سازگار با آن يك لبه با وزن $-w_{ob}$ قرار مي‌دهيم. اين امر به اين معني است كه ما ترجيح مي‌دهيم براي افزايش قابليت كنترل (مشاهده) متغيرهاي مياني و متغيرهاي PI (PO) را بر روي يك ثبات نگاشت كنيم. تمام لبه‌هاي ديگر داراي وزن صفر هستند.

كاهش عمق ترتيبي و حلقه‌ي ترتيبي: براي تمام زوج متغيرهاي سازگار(v_i, v_j) كه بر روي زوج ثبات (R_i, R_j) نگاشت مي‌شوند، دو ثبات را با هم ادغام كرده و DPCG مدار مربوطه را مي‌سازيم و از روي آن پارامترهاي زير را محاسبه مي‌كنيم:

- P_{ij} : طول بيشينه‌ي كوتاه‌ترين مسير موجود بين ورودي‌هاي اصلي و خروجي‌هاي اصلي
- c_{ij} : طول بزرگ‌ترين چرخه
- n_{ij} : تعداد چرخه‌هاي مستقل

از آنجا كه عمق ترتيبي به‌صورت خطي بر روي زمان ATPG تاثير مي‌گذارد، بين هر دو گره‌ي v_i و v_j يك لبه با وزن $(P_{ij} - P_0)$ قرار مي‌دهيم. در اين رابطه P_0 نشان‌دهنده‌ي عمق ترتيبي در مدار اوليه (يعني پيش از ادغام متغيرهاي v_i و v_j) است. اگر P_{ij} از P_0 كمتر باشد، به اين معني است كه عمق ترتيبي مدار پس از ادغام متغيرهاي v_i و v_j كمتر شده است، در اين‌صورت مقدار وزن منفي مي‌شود كه تمايل به ادغام اين دو ثبات را نشان مي‌دهد. طول بزرگ‌ترين چرخه به‌صورت نمايي بر روي زمان ATPG تاثير مي‌گذارد، از اين رو بين هر دو متغير v_i و v_j يك لبه با وزن $2^{c_{ij}}$ قرار مي‌دهيم. اين وزن نشان‌دهنده‌ي عدم تمايل ما به ادغام اين دو ثبات است. مي‌توان با قرار دادن يكي از ثبات‌هاي يك چرخه‌ي مستقل بر روي مسير پويش[۳۱] ، چرخه را شكست. از اين رو تعداد چرخه‌هاي مستقل نشان‌دهنده‌ي كمترين تعداد ثبات‌هايي است كه بايد بر روي مسير پويش قرار بگيرند. بين هر دو متغير v_i و v_j يك لبه با وزن n_{ij} قرار مي‌دهيم كه نشان مي‌دهد اگر ادغام دو متغير باعث افزايش تعداد چرخه‌هاي مستقل شود، به همان نسبت تمايل ما به اين ادغام كمتر خواهد شد.

حذف حلقه به خود: اگر ثبات خروجي و يكي از ثبات‌هاي ورودي يك پيمانه بر روي يك ثبات نگاشت شوند، در مدار نهايي يك ثبات خود مجاور ايجاد خواهد شد. براي حذف اين ثبات، يك لبه با وزن $+w_{self}$ بين هر دو زوج متغير ورودي و خروجي هر پيمانه قرار مي‌دهيم. به اين ترتيب تلاش مي‌كنيم كه متغيرهاي ورودي و خروجي يك پيمانه بر روي ثبات‌هاي جداگانه نگاشت شوند. تمام لبه‌هاي ديگر داراي وزن صفر هستند.

۳-۳ پالايش گراف ECG

در مرحله‌ي پالايش گراف ECG ، تمام لبه‌هاي موجود بين دو گره با يك لبه جايگزين مي‌شود. وزن اين لبه برابر حاصل‌جمع وزن‌دار وزن‌هاي لبه‌هاي موجود بين اين دو گره است. بنابراين وزن اين لبه كه نشان‌دهنده‌ي هزينه‌ي ادغام اين دو متغير است، با رابطه‌ي زير بيان مي‌شود.

$$w_{ij} = -\alpha_1 w_{co} - \alpha_2 w_{ob} + \alpha_3 (P_{ij} - P_0) + \alpha_4 2^{c_{ij}} + \alpha_5 n_{ij} + \alpha_6 w_{self} \qquad (۱)$$

در اين رابطه ضرايب α_1، α_2 ، α_3 ، α_4 ، α_5 و α_6 ضرايب مورد نظر براي تعيين اهميت اهداف بهينه‌سازي هستند كه توسط كاربر تعريف مي‌شوند. براي مثال اگر اهميت كاهش زمان ATPG از ساير اهداف بيشتر باشد، بايد مقدار ضرايب α_3، α_4 و α_5 بيشتر از ساير ضرايب باشد.

در مقداردهي به ضرايب بايد به اين نكته توجه داشت كه نسبت مقادير ضرايب است كه درجه‌ي اهميت يكي از اهداف را نسبت به اهداف ديگر تأييد مي‌كند. بنابراين مقدار يك ضريب به تنهايي هيچ‌گونه معياري براي افزايش يا كاهش اهميت يك هدف محسوب نمي‌شود.

۴-۳ الگوريتم تخصيص ثبات

الگوريتم تخصيص ثبات ما در شكل ۵ نشان داده شده است. در ابتدا با اجراي الگوريتم حريصانه‌ي LEA كم‌ترين تعداد ثبات مورد نياز براي پياده‌سازي مدار را به‌دست مي‌آوريم. براي افزايش قابليت كنترل بين هر متغير مياني و هر PI يك لبه با وزن $-w_{co}$ و نيز براي افزايش قابليت مشاهده بين هر متغير مياني و هر PO يك لبه با وزن $-w_{ob}$ قرار مي‌دهيم. براي كاهش زمان ATPG، براي هر زوج متغير سازگار ، ابتدا دو متغير (u, v) را بر روي يك ثبات نگاشت مي‌كنيم و

```
Testable_Register_Allocation(){
   1. min_reg = Greedy LEA
        2. For each intermediate variable u add an
      edge between u and PI with specified
      weigh (-w_co);
   3. For each intermediate variable u add an
      edge between u and PO with specified
      weigh (-w_ob);
   4. For all compatible variable pairs of (u, v)
      add three edges between u and v
      with weights (P_uv-P_0), 2^c_uv, n_uv;
   5. Add an edge with specified
      weigh (+w_self) between the inputs
      and the output variables of a module;
   6. Refine ECG;
   7. Simulated_Annealing_Graph_Coloring(min_reg);
 }
```

شکل ۵- الگوريتم تخصيص ثبات آزمون‌پذير

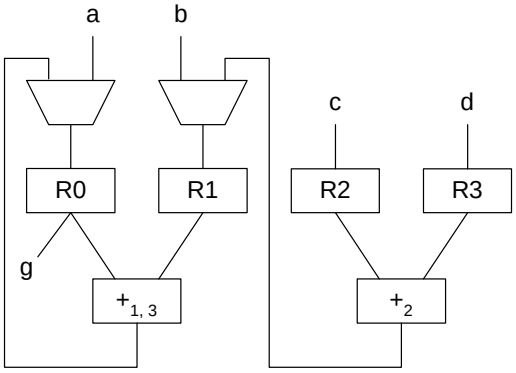
سپس سه لبه با وزن‌هاي $(P_{uv} - P_0)$، $2^{c_{uv}}$ و n_{uv} بين دو گره‌ي u و v اضافه مي‌كنيم.

براي حذف حلقه به خود يك لبه با وزن $w_{self}+$ بين متغيرهاي ورودي و خروجي يك پيمانه اضافه مي‌شود. سپس عمليات پالايش گراف ECG انجام مي‌شود و در پايان براي تخصيص ثبات با هدف افزايش آزمون‌پذيري با كمترين تعداد ثبات، الگوريتم رنگ‌آميزي گراف وزن‌دار بر روي گراف ECG پالايش شده اجرا مي‌شود. روش مورد استفاده براي رنگ‌آميزي گراف وزن‌دار مبتني بر يك روش جستجوي مكاشفه‌اي به نام *Simulated Annealing* است. اين الگوريتم بر اين اصل استوار است كه اگر براي يك مسئله يك پاسخ داشته باشيم، مي‌توانيم براي رسيدن به يك پاسخ بهتر در همسايگي پاسخ موجود جستجو كنيم. در صورت پيدا شدن يك پاسخ جديد، پاسخ فعلي را با پاسخ جديد جايگزين مي‌كنيم و در غير اين‌صورت پاسخ موجود را به‌صورت احتمالاتي تغيير مي‌دهيم.

براي مثال مدار *ex1* را در نظر بگيريد. با اعمال روش زمان‌بندي نيروگرا[۳۲] عمليات $+_1$ و $+_2$ در زمان ۰ و عمليات $+_3$ در زمان ۱ زمان‌بندي مي‌شود. با استفاده از الگوريتم LEA براي تخصيص پيمانه، به نتيجه‌ي $M_{LEA}=\{(+_{1,3}),(+_2)\}$ خواهيم رسيد كه نشان مي‌دهد عمليات $+_1$ و $+_3$ بر روي يك جمع‌كننده و عمليات $+_2$ بر روي جمع‌كننده‌ي ديگري نگاشت شده است. استفاده از الگوريتم LEA براي تخصيص ثبات منجر به نتيجه‌ي $R_{LEA}=\{(a,e,g),(b,f),(c),(d)\}$ مي‌شود. مدار به دست آمده از روش تخصيص ثبات LEA در شكل ۶ نمايش داده شده است.

حال فرض كنيد كه براي همين مدار با فرض تخصيص پيمانه مشابه حالت قبل، از روش تخصيص ثبات ارائه شده در اين مقاله استفاده كنيم. در اين آزمايش مقادير w_{co} ، w_{ob} و w_{self} را برابر ۱ فرض مي‌كنيم. هم‌چنين فرض مي‌كنيم كه هدف ما از سنتز آزمون‌پذير بهبود قابليت كنترل/مشاهده و حذف حلقه به خود باشد. در اين‌صورت مقادير ضرايب برابر $\alpha_1 = 10, \alpha_2 =10, \alpha_3 = 0, \alpha_4 = 0, \alpha_5 = 0, \alpha_6 = 10$ خواهد بود. در اولين مرحله بايد گرافECG را ايجاد كنيم، براي اين منظور با توجه به مقادير ضرايب بايد مراحل ۱، ۳ و ۵ از الگوريتم را انجام دهيم. پس از آن مرحله‌ي پالايش گراف ECG انجام مي‌شود (مرحله‌ي ۶). در پايان، الگوريتم رنگ آميزي گراف وزن‌دار بر روي گراف پالايش شده‌ي ECG انجام مي‌شود (مرحله‌ي ۷). شكل ۷ گراف ECG را در هر مرحله به صورت ماتريس همسايگي نشان مي‌دهد. انتخاب اين روش براي تخصيص ثبات منجر به نتيجه‌ي $R_{WCG}=\{(b,f,g),(d),(c,e),(a)\}$ خواهد شد. شكل ۸ مدار حاصل از روش تخصيص ثبات WGC را نشان مي‌دهد.

براي مقايسه‌ي نتايج حاصل از R_{LEA} و R_{WCG} با يكديگر، هر دو مدار حاصل را (با عرض گذرگاه ۴ بيتي) به يك ابزار سنتز سطح RTL اعمال كرديم.



شكل ۶- پياده‌سازي RTL مدار ex1 با استفاده از روش تخصيص ثبات LEA

	a	b	c	d	e	f	g
a	0	*	*	*	-1	-1	0
b	*	0	*	*	-1	-1	0
c	*	*	0	*	-1	-1	0
d	*	*	*	0	-1	-1	0
e	-1	-1	-1	-1	0	*	0
f	-1	-1	-1	-1	*	0	0
g	0	0	0	0	0	0	0

(الف)

	a	b	c	d	e	f	g
a	0	*	*	*	0	0	-1
b	*	0	*	*	0	0	-1
c	*	*	0	*	0	0	-1
d	*	*	*	0	0	0	-1
e	0	0	0	0	0	*	-1
f	0	0	0	0	*	0	-1
g	-1	-1	-1	-1	-1	-1	0

(ب)

	a	b	c	d	e	f	g
a	0	*	*	*	+1	0	0
b	*	0	*	*	+1	0	0
c	*	*	0	*	0	+1	0
d	*	*	*	0	0	+1	0
e	+1	+1	0	0	0	*	+1
f	0	0	+1	+1	*	0	+1
g	0	0	0	0	+1	+1	0

(ج)

	a	b	c	d	e	f	g
a	0	*	*	*	0	-10	-10
b	*	0	*	*	0	-10	-10
c	*	*	0	*	-10	0	-10
d	*	*	*	0	-10	0	-10
e	0	0	-10	-10	0	*	0
f	-10	-10	0	0	*	0	0
g	-10	-10	-10	-10	0	0	0

(د)

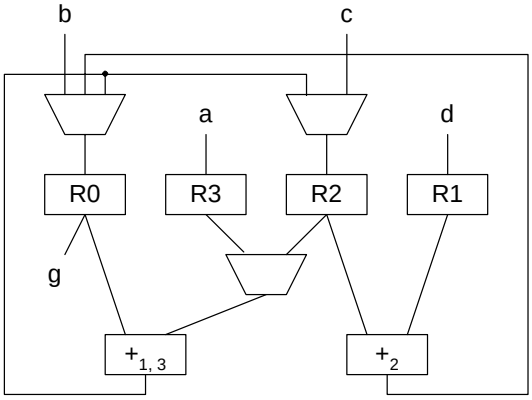
شكل ۷- (الف) گراف ECG براي بهبود قابليت كنترل، (ب) گراف ECG براي بهبود قابليت مشاهده، (ج) گراف ECG براي كاهش حلقه به خود، (د) گراف ECG پالايش شده

خروجي حاصل از ابزار سنتز به يك ابزار ATPG در سطح گيت به نام HITEC PROOFS- اعمال شد [۲۱]. نتايج حاصل از اين شبيه‌سازي در قبال جريمه‌ي ناچيز تاخير، بهبود قابل توجهي را در پارامترهاي آزمون‌پذيري نشان مي‌دهد. زمان ATPG، پوشش خطا و تاخير براي روش‌هاي R_{LEA} و R_{WCG} به ترتيب برابر ۲/۳۷ ثانيه، ۹۸%، ۳/۴ نانوثانيه، ۰/۲۶ ثانيه، ۱۰۰% و ۴/۵ نانوثانيه خواهد بود.هم‌چنين خروجي حاصل از ابزار سنتز را با يك مجموعه‌ي بردار آزمون تصادفي به يك ابزار شبيه ساز خطا به نام HITEC-PROOFS اعمال كرديم.

نتايج حاصل نشان مي‌دهد پوشش خطاي به‌دست آمده براي دو روش R_{LEA} و R_{WGC} به ترتيب برابر ۸۶% و ۱۰۰% است، كه نشان مي‌دهد روش R_{WGC} به مداري منتهي مي‌شود كه طبيعتا آزمون‌پذيرتر است.

۴- نتایج آزمایش‌ها

براي ارزيابي روش تخصيص ثبات ارائه شده، هفت مدار مختلف انتخاب شده است. مدار اول كه آن را *cir1* مي‌ناميم بخشي از افزارسنجی[۳۳] *DiffEq* است [۲۵]. مدارهاي ديگر عبارتند از: *DiffEq* و *Paulin* از مرجع[۲۲]. مدار *oven-ctrl* از مرجع [۲۶], مدار *real* از مرجع[۲۲], مدار *TsengA* از مرجع [۲۷] و مدار *EWF* از مرجع [۷]. پهناي گذرگاه تمام مدارهاي انتخاب شده ۸ بيت است.



شكل ۸- پيادەسازي RTL مدار *ex1* با استفاده از روش تخصيص ثبات WGC

در ابتدا با استفاده از روش پيشنهادي خود، سنتز مسير داده‌ي هريك از اين مدارها را انجام مي‌دهيم و كد Verilog مسير داده را توليد كرده، سپس آن را به سطح گيت تبديل مي‌كنيم. پس از آن خروجي حاصل را به قالب ISCAS تبديل كرده و آن را به ابزار HITEC-PROOFS اعمال مي‌كنيم تا پارامترهاي آزمون‌پذيري مدار را محاسبه كند. در انجام آزمايش‌ها فرض شده است كه مقادير w_{co} ، w_{ob} و w_{self} برابر ۱ هستند. هم‌چنين فرض شده است كه تمام ضرايب α_1 تا α_6 برابر ۱۰ هستند كه نشان مي‌دهد تمام معيارهاي آزمون‌پذيري داراي اهميت يكساني هستند. نتايج حاصل از آزمايش‌ها در جدول ۱ خلاصه شده است. كارايي خطا[۳۴] نشان‌دهنده‌ي نسبت تعداد خطاهاي تشخيص داده شده به كل خطاهاي موجود در مدار است. زمان ATPG نشان‌دهنده‌ي مدت زمان صرف شده براي توليد بردارهاي آزمون است. ستون آخر جدول پوشش خطا را با اعمال بردارهاي آزمون تصادفي نشان مي‌دهد. اين ستون نشان‌دهنده‌ي ميزان سادگي آزمون مدار است.

همان‌طور كه در جدول ۱ مشاهده مي‌شود، استفاده از روش تخصيص ثبات پيشنهادي ما:

- زمان ATPG را به‌صورت قابل توجهي كاهش مي‌دهد
- تعداد خطاهاي عقيم‌مانده[۳۵] را كاهش داده و در نتيجه كارايي خطا را افزايش مي‌دهد
- سبب افزايش پوشش خطا مي‌شود

شكل ۹ اين نتايج را به صورت گرافيكي نمايش مي دهد.

مقايسه‌اي بين روش تخصيص ثبات PHITS [۶] و روش ارائه شده در اين مقاله در جدول ۲ آمده است. براي مقايسه‌ي اين دو روش فرض شده است كه پهناي گذرگاه مدارهاي انتخابي برابر ۲ بيت است. همان‌طور كه در اين جدول مشاهده مي‌شود، نتايج به دست آمده از روش تخصيص ثبات WGC در تمام موارد در مقايسه با روش PHITS بهبود قابل ملاحظه‌اي را در زمان ATPG نشان مي‌دهد. هم‌چنين پوشش خطا نيز در تمام موارد به به جز در مدار *DiffEq* بهتر شده است. دليل اين امر اين است كه روش پيشنهادي در اين مقاله براي سنتز مسير داده[۳۶] ارائه شده است. از اين‌رو در مدارهايي مانند *DiffEq* كه داراي عملگرهايي با طبيعت كنترلي مانند مقايسه‌كننده هستند، لزوما پوشش خطاي بهتري به دست نخواهد آمد. بررسي نتايج مدار *Cir1* كه از حذف بخش حاوي عملگر مقايسه‌كننده از مدار *DiffEq* به دست آمده، تاييد كننده‌ي اين امر است.

۵- نتيجه‌گيري

در اين مقاله در نظر گرفتن ملاحظات آزمون‌پذيري در طي عمليات تخصيص ثبات مورد بررسي قرار گرفت. براي اين‌منظور در تخصيص ثبات بهبود برخي معيارهاي آزمون‌پذيري سطح بالا مانند بهبود قابليت كنترل/مشاهده، كاهش عمق ترتيبي، كاهش طول حلقه‌هاي ترتيبي و حذف حلقه‌هاي به خود مورد توجه قرار مي‌گيرد كه نتيجه‌ي آن كاهش زمان ATPG و افزايش پوشش خطا مي‌باشد. در اين مقاله ما يك روش جديد تخصيص ثبات معرفي كرديم كه مبتني بر يك مدل مجرد گراف با نام ECG است. برخلاف روش‌هاي ديگر كه يك يا حداكثر دو معيار آزمون‌پذيري را در طي مراحل سنتز در نظر مي‌گيرند، روش پيشنهادي ما با هدف بهبود هم‌زمان تمام معيارهاي آزمون‌پذيري در طي مرحله‌ي تخصيص ثبات ارائه شده است. نتايج حاصل از آزمايش‌ها نشان مي‌دهد كه روش پيشنهادي ما با مقدار كمي افزايش تاخير مدار، به بهبود قابل توجهي در زمان ATPG و پوشش خطاي مدار مي‌رسد.

مراجع

[1] M. C. McFarland, A. C. Parker, and R. Camposano, "Tutorial on High-Level Synthesis," *Proc. of 25th Design Automation Conf.*, pp. 330-336, California, 1988.

[2] M. C. McFarland, A. C. Parker, and R. Camposano, "The High-Level Synthesis of Digital Systems," *Proc. of the IEEE*, vol. 78, no. 2, pp. 310-318, 1990.

[3] M .L. Bushnell, and V. D. Agrawal, *Essentials of Electronic Testing for Digital, Memory and Mixed-Signal VLSI Circuits*, Kluwer Academic Publishers, 2000.

[4] S. Dey, A. Raghunatan, and R. K. Roy, "Considering Testability During High-Level Design," *ASP DAC*, pp. 205-210, 1998.

[5] L. J. Avra, and E.J. McCluskey, "High-level Synthesis of Testable Design: An Overview of University systems," *IEEE Intl Test Conf –Test Synthesis Seminar*, 1994.

[6] D. Wagner, and S. Dey, "High-Level Synthesis for Testability: a Survey and Perspective", *Proc. of Design Automation Conf.*, pp.131-136, 1996.

[7] M. Tien, and C. Lee, *High Level Test Synthesis of Digital VLSI Circuits, Artech House Press*, 1997.

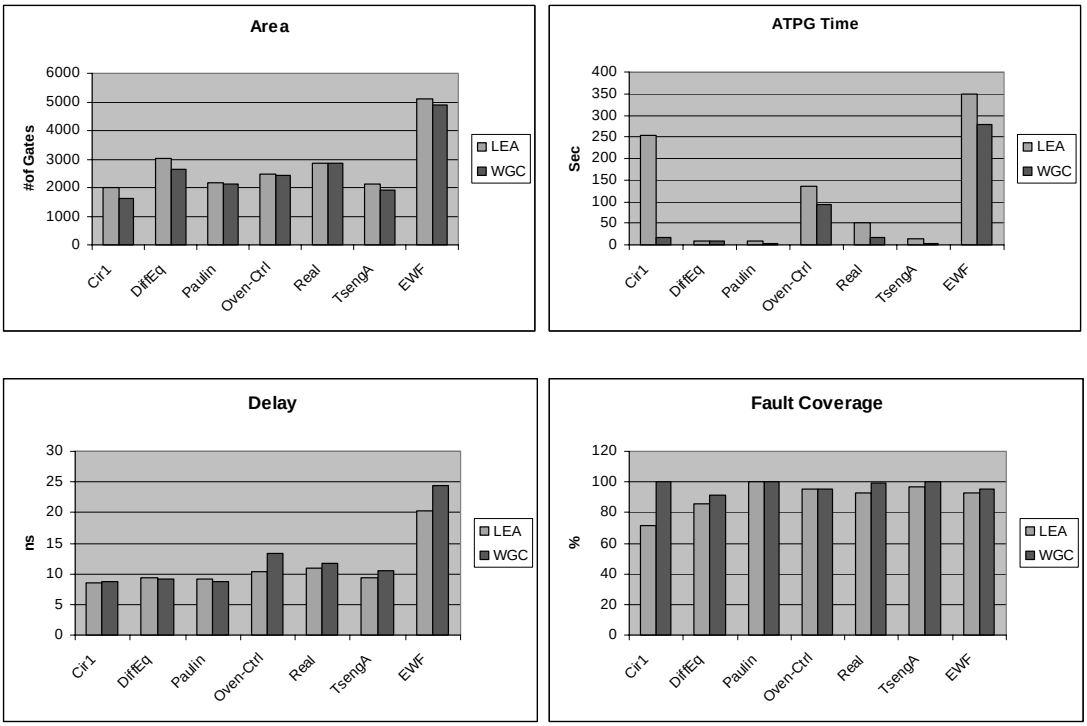
[8] S. K. Chiu, and C. Papachristou, "A Built-In Self-Testing Approach for Minimizing Hardware Overhead," *Proc. of Intl Conf. Computer Design*, pp. 282-285, 1991.

جدول ۱- نتایج آزمایش‌ها

Circuit	Allocation Method	Module Allocation	Register Allocation	Run Time (sec)	Area (#gates)	Delay (ns)	ATPG			Fault Coverage (RTP)
							Fault Coverage	Fault Efficiency	ATPG Time (sec)	
Cir1	LEA	$(*_{1,4,5})$, $(*_{2,3})$, $(-_{1,2})$	R0=(u, e, u1), R1=(z, b, c), R2=(y, d), R3=(A), R4=(dz, f)	0.58	1998	8.5	71.8%	96%	235.33	62.7%
	WGC	$(*_{1,4,5})$, $(*_{2,3})$, $(-_{1,2})$	R0=(y), R1=(z, b, d, e, u1), R2=(a, c), R2=(dz, f), R4=(u)	0.82	1605	8.7	100%	100%	16.733	94.1%
DiffEq	LEA	$(*_{1,3,5})$, $(*_{2,4,6})$, $(+_{1,2})$, $(-_{1,2})$, $(<_1)$	R0=(u, e, u1), R1=(dz, f, y1), R2=(z, a, c, g), R3=(y), R4=(A, d), R5=(b, ctrl1), R6=(z1)	0.7	3020	9.4	85.9%	98.6%	9.267	83%
	WGC	$(*_{1,3,5})$, $(*_{2,4,6})$, $(+_{1,2})$, $(-_{1,2})$, $(<_1)$	R0=(b, ctrl1), R1=(dz, g), R2=(z1), R3=(z, a, c, f, y1), R4=(A, d, e, u1), R5=(y), R6=(u)	0.93	2628	9.2	91.3%	99.9%	8.5	87%
Paulin	LEA	$(*_{1,3,5})$,$(*_{2,4,6})$, $(+_{1,2})$, $(-_{1,2})$	R0=(u, t6, Z), R1=(dx, t7, Y), R2=(x, t1, t4, t8), R3=(y), R4=(t2, t5), R5=(X)	0.6	2154	9.2	99.9%	99.9%	8.23	93%
	WGC	$(*_{1,3,5})$,$(*_{2,4,6})$, $(+_{1,2})$, $(-_{1,2})$	R0=(y), R1=(x, t1, t4, t7, Y), R2=(u), R3=(X), R4=(dx, t6), R5=(t2, t5, t8, Z)	0.7	2134	8.8	100%	100%	3.7	97%
Oven-Ctrl	LEA	$(+_{1,2,3,5})$, $(+_4)$, $(*_1)$, $(/_1)$, $(-_1)$	R0=(Tset, TDIFF), R1=(T4, a, TERR), R2=(T1, c), R3=(T2, TAVG), R4=(T3), R5= (b, d, e)	0.4	2478	10.4	95.3%	99.6%	136.11	67.9%
	WGC	$(+_{1,2,3,5})$, $(+_4)$, $(*_1)$, $(/_1)$, $(-_1)$	R0=(b, d, c, e, TDIFF), R1=(T2, TAVG), R2=(Tset), R3=(T4, a, TERR), R4=(T1), R5= (T3)	0.9	2426	13.3	95.3%	99.9%	94.28	75%
Real	LEA	$(*_{1,2,3,4})$, $(+_{1,2,3})$, $(-_{1,2})$, $(/_1)$, $(/_2)$	R0=(b, g, h, j, X1), R1=(c, d, I, k, X2), R2=(a, l), R3=(e,f)	0.5	2870	10.9	92.6%	97%	49.65	84.2%
	WGC	$(*_{1,2,3,4})$, $(+_{1,2,3})$, $(-_{1,2})$, $(/_1)$, $(/_2)$	R0=(d), R1=(a, l), R2=(c, e, f, I, k, X1), R3=(b, g, h, j, X2)	0.8	2856	11.8	99.2%	100%	17.8	98%
TsengA	LEA	$(*_{1,2})$,$(+_{1,2})$, $(+_3)$, $(-_1)$, $(\|_1)$, $(\&_1)$	R0=(v1, b, d, W3), R1=(v2, W2), R2=(v3, a, e), R3=(v4, c, f), R4=(v5)	0.8	2127	9.4	97.3%	98.4%	13.3	84.3%
	WGC	$(*_{1,2})$,$(+_{1,2})$, $(+_3)$, $(-_1)$, $(\|_1)$, $(\&_1)$	R0=(v3, a), R1=(v4, b, f, W3), R2=(v5,e), R3=(v1, c, d, W2), R4=(v2)	0.95	1931	10.6	99.9%	100%	3.8	88.7%
EWF	LEA	$(*_{1,2,3,4,5,6,7,8})$, $(+_{1,3,5,7,8,9,11,12,14,16,18,22,25})$, $(+_{2,4,6,10,15,17,19,21,23,24})$, $(+_{13,20,26})$	R0=(T_{26},x_1,e,f,x_{10},OUT,t_{39}), R1=(IN,x_{13},k,t_2), R2=(T_2,a,x_7, h,x_{15}), R3=(T_{13},x_{14},t_{13}), R4=(T_{38},t_{18}), R5=(X_{15},t_{38},b,x_5, x_9,x_4,x_8,x_{11},x_{16}), R6=(K,t_{33},g, t_{26}), R7=(t_{39},j), R8=(T_{18}), R9=(x_3,x_2,x_6,c), R10=(d,x_{12})	2.3	5107	20.18	93%	96%	349	81%
	WGC	$(*_{1,2,3,4,5,6,7,8})$, $(+_{1,3,5,7,8,9,11,12,14,16,18,22,25})$, $(+_{2,4,6,10,15,17,19,21,23,24})$, $(+_{13,20,26})$	R0=(t_2,g,c), R1=(t_{26},x_3,f), R2=(t_{13},x_2,d,x_7,j), R3=(x_{15},t_{33}, x_1,e,x_4,x_6,x_5,x_{12},h), R4=(a,x_8, x_{11},x_{13},OUT), R5=(k,b,x_{10}), R6=(t_{18}), R7=(t_{38}),R8=(t_{39},x_{14}), R9=(IN,x_{16}), R10=(x_9)	9.6	4876	24.5	95.3%	97%	278	84%

جدول ۲- مقایسه‌ي بین زمان تولید بردار آزمون و پوشش خطا در دو روش تخصیص ثبات PHITS و WGC

Circuit	Allocation Method	ATPG Time (sec)	Fault Coverage
Cir1	PHITS	94.2	98.69%
	WGC	2.1	100%
DiffEq	PHITS	709	99.78%
	WGC	2.3	96.7%
Oven-Ctrl	PHITS	665.3	93.98%
	WGC	2.38	99.7
Real	PHITS	399.5	97.96%
	WGC	2.65	98%
TsengA	PHITS	426.8	99.21%
	WGC	1.3	100%



شکل ۹- مساحت، تاخیر، زمان ATPG و پوشش خطا

[14] F. F. Hsu, and J. H. Patel, "High-Level Variable Selection for Partial-Scan Implementation," *Proc. of the IEEE Intl Conf. on Computer Aided Design*, pp. 79-84, 1998.

[15] M.L. Flottes, R. Pires, B. Rouzeyre, and L. Volpe, "Scanning Data Paths: A Fast and Effective Partial Scan Technique," *Proc. of DATE*, pp. 158-165, 1998.

[16] H. C. Tsai, K. T. Cheng, and S. Bhawmik, "On Improving Test Quality of Scan-based BIST," *IEEE Trans*. On CAD, vol. 19, no. 8, pp. 928-938, 2000.

[17] S. Ravi, G. Lakshminarayana, and N. K. Jha, " TAO: Regular Expression Based Register-transfer Level Testability Analysis and Optimization," *IEEE Trans*. On VLSI Systems, vol. 9, pp 824-832, Dec. 2001.

[18] D. Xiang and H. Fujiwara, "Handling the Pin Overhead Problem of DFTs for High-quality and At-speed Test," *IEEE Trans. On CAD*, vol. 21, no. 9, pp. 1105-1113, 2002.

[9] L. Avra, "Allocation and Assignment in High-Level Synthesis for Self-Testable Data Paths," *Proc. of Intl Test Conf.*, pp. 463-471, 1991.

[10] S. Dey, M. Potkonjak, and R. K. Roy, "Exploiting Hardware Sharing in High-Level Synthesis for Partial Scan Optimization," *Proc. of the IEEE Intl Conf. on Computer Aided Design*, pp. 20-25, 1993.

[11] A. Majumdar, K. Saluja, and R. Jain, "Incorporating Testability Considerations in High-Level Synthesis," *Proc. of Intl Symp. Fault-Tolerant Computers*, pp. 272-279, 1992.

[12] T. Lee, N.K. Jha, and W.H. Wolf, "Behavioral Synthesis of Highly Testable Data Paths under the Non-Scan and Partial Scan Environments," *Proc. of Design Automation Conf.*, pp. 292-297, 1993.

[13] J. Steensma, F. Catthor, and H.D. Man, " Test of high Throughput Data Paths with Symbolic Controllability and Observability Descriptions," *In Proc. of* 6th *Workshop New Directions for Testing*, pp 67-76, 1992.

[19] T. Yang, and Z. Peng, "An Efficient Algorithm to Integrated Scheduling and Allocation in High-Level Test Synthesis," *Proc. of DATE*, pp. 74-81, 1998.

[20] R. Camposano, and W. Rosenstiel, "Synthesizing Circuits from Behavioral Descriptions," *IEEE Trans. on CAD/ICAS*, vol. 8, no. 2, pp. 171-180 1989.

[21] A. Hashimoto, and J. Stevens, "Wire Routing by Channel Assignment within Large Apertures," *Proc. of Design Automation Workshop*, pp. 155-169, 1971.

[22] F. J. Kurdahi, and A.C. Parker, "REAL: A Program for Register Allocation," *Proc. of Design Automation Conf.*, pp. 210-215, 1987.

[23] A. Steininger, "Testing and Built-In Self-Test – A Survey," *Journal of Systems Architecture*, vol. 46, pp. 721-747, 2000.

[24] T. Niermann, and J. H. Patel, "HITEC: A Test Generation Package for Sequential Circuits," *Proc. of the European Conf. on Design Automation*, pp. 214-218, 1992.

[25] P.G. Paulin, J.P. Knight, and E.F. Girczyc, " HAL: A Multi-Paradigm Approach to Automatic Data Path Synthesis," *Proc. of Design Automation Conf.*, pp. 263-270, 1989.

[26] E. F. Girczyc, R. J. Buhr, and J. P. Knight, "Application of a Subset of ADA as an Algorithmic Hardware Description Language for Graph-Based Hardware Compilation," *IEEE Trans. on Computer Aided Design*, vol. 4, pp. 134-142, 1985.

[27] C. J. Tseng, and D. P. Siewiorek, "Automated Synthesis of Data-Paths in Digital Systems," *IEEE Trans. on Computer Aided Design*, vol. 5, pp. 379-395, 1986.

[1] CAD Tool
[2] Behavioral Synthesis Tool
[3] Register Transfer Level (RTL)
[4] Behavioral Testable Synthesis
[5] Automatic Test Pattern Generation
[6] Built-In Self Test
[7] Sequential Depth
[8] Sequential Loop
[9] Controllability/Observability
[10] Self-Loop
[11] Hard to Test
[12] Data Flow Graph (DFG)
[13] Scheduled Data Flow Graph
[14] Left-Edge Algorithm
[15] Clique Partitioning
[16] Graph Coloring
[17] Compatibility Graph
[18] Conflict Graph
[19] Primary Input (PI)
[20] Primary Output (PO)
[21] Data Path Circuit Graph (DPCG)
[22] Non-Self-Loop
[23] Built-In Logic Block Observer
[24] Serial Scan In/Out
[25] Random Test Pattern Generation (RTPG)
[26] Multiple-Input Signature Register
[27] Self-Adjacent
[28] Concurrent BILBO
[29] Extended Conflict Graph (ECG)
[30] Structural Constraint
[31] Scan Path
[32] Force-Directed Scheduling
[33] Benchmark
[34] Fault Efficiency
[35] Aborted Faults
[36] Data-Path Synthesis

سعید صفري تحصيلات خود را در مقاطع كارشناسي و كارشناسي ارشد مهندسي كامپيوتر به ترتيب در سالهاي ۱۳۷۳ و ۱۳۷۶ در دانشگاه صنعتي شريف و دانشگاه تهران به پايان رساند و هم‌اكنون در دانشگاه صنعتي شريف در مقطع دكتراي كامپيوتر مشغول به تحصيل مي‌باشد. زمينه‌هاي تحقيقاتي مورد علاقه‌ي ايشان عبارتند از: سنتز مدارهاي ديجيتال، آزمون مدارهاي ديجيتال و طراحي ابزارهاي طراحي سخت‌افزار به كمك كامپيوتر. آدرس پست الكترونيكي نامبرده عبارتست از:

safari@ mehr.sharif.edu

امیرحسین جهانگیر كارشناسي ارشد مهندسي برق خود را در سال ۱۳۶۳ از انستيتوي ملي علوم كاربردي و كارشناسي ارشد اتوماتيك و انفورماتيك خود را در سال ۱۳۶۴ از دانشگاه كلودبرنارد ليون و دكتراي خود را در انفورماتيك صنعتي در زمينه‌ي ستابدهي سخت‌افزاري شبيه سازمنطقي مبتني بر رويداد در انستيتوي ملي علوم كاربردي تولوز فرانسه اخذ نمود. وي از سال ۱۳۶۸ تاكنون استاديار دانشكده‌ي مهندسي كامپيوتر دانشگاه صنعتي شريف است. زمينه‌هاي تحقيقاتي مورد علاقه‌ي ايشان پردازش موازي، شبكه‌هاي كامپيوتري و اتوماسيون صنعتي است. ايشان تاكنون مسئوليت‌هاي اجرايي مختلفي چون رياست دانشكده‌ي مهندسي كامپيوتر دانشگاه صنعتي شريف و رياست مركز محاسبات دانشگاه صنعتي شريف را بر عهده داشته‌اند. آدرس پست الكترونيكي نامبرده عبارتست از:

jahangir@sharif.edu

هادي اسمعيل‌زاده درجه‌ي كارشناسي خود را در سال ۱۳۸۱ از دانشگاه تهران با عنوان دانشجوي ممتاز در رشته‌ي مهندسي كامپيوتر دريافت كرد. او در حال حاضر دانشجوي كارشناسي ارشد رشته‌ي معماري كامپيوتر در دانشگاه تهران مي‌باشد. علايق تحقيقاتي او شامل طراحي سخت‌افزار با الهام زيستي، شبكه‌هاي عصبي، نظريه‌ي يادگيري، جنبه‌هاي مختلف طراحي SOC و آزمون سيستم‌هاي ديجيتال مي‌باشد. آدرس پست الكترونيكي نامبرده عبارتست از:

hadi@cad.ece.ut.ac.ir