افزایش تحمل‌پذیری مدارهای ترکیبی در مقابل خطای نرم مبتنی بر جایگزینی منطقی محلی

محمدرضا روحانی پور[1]، بهنام قوامی[2]، محسن راجی[3]
[1]دانشجوی کارشناسی ارشد معماری کامپیوتر، دانشگاه شهید باهنر کرمان
[2]استادیار گروه مهندسی کامپیوتر، دانشکده فنی و مهندسی، دانشگاه شهید باهنر کرمان، ghavami@uk.ac.ir
[3] استادیار دانشکده برق و کامپیوتر، دانشگاه شیراز

چکیده

*در این مقاله یک تکنیک باز-سنتز مبتنی بر جایگزینی منطقی محلی به‌منظور کاهش نرخ خطای نرم مدارهای ترکیبی ارائه شده است. روش پیشنهادی راه‌کار خلاقانه‌ای را جهت افزایش احتمال پوشش منطقی مدار تحت خطای نرم و با در نظر گرفتن محدودیت مساحت فراهم می‌آورد. در این تکنیک، ابتدا زیرمداری از مدار اصلی استخراج می‌شود و به کمک روش **Quine–McCluskey (QM)** توسعه‌یافته، پیاده‌سازی‌های مختلف از زیرمدار به دست می‌آید. به‌منظور انتخاب بهترین جایگزین از بین مجموعه پیاده‌سازی‌های مختلف زیرمدار با کمترین نرخ خطای نرم، معیار جدید **Global Failure Probability (GFP)** ارائه شده است. این معیار، با در نظر گرفتن پارامترهای نظیر احتمال ایجاد پالس‌های گذرای خطادار در زیرمدار و احتمال انتشار این پالس‌ها توسط زیرمدار، این امکان را فراهم می‌آورد که پس از تغییر محلی ساختار زیرمدار بتوان به ارزیابی سراسری نرخ خطای نرم ناشی از این تغییر پرداخت. به‌عبارت‌دیگر، معیار ارائه‌شده در هر مرحله از فرآیند بهبود نرخ خطای نرم بدون نیاز به جایگزین نمودن تمام پیاده‌سازی‌های مختلف از زیرمدار در مدار اصلی و محاسبه نرخ خطای نرم کل مدار به ازای هرکدام از آن‌ها، تاثیر هر یک را در نرخ خطای نرم مدار اصلی محاسبه می‌نماید. نتایج حاصل از شبیه‌سازی‌ها بر روی مدارهای محک **ISCAS'85** نشان می‌دهد که روش پیشنهادی به‌طور میانگین موجب ۱۷/۷۵٪ کاهش نرخ خطای نرم به ازای ۵/۳۹٪ سربار مساحت شده است.*

کلید واژه

خطای نرم، زیرمدار، پوشش منطقی، مدار ترکیبی، نرخ خطای نرم.

مقدمه

نیاز به کارایی بیشتر، طراحان مدارهای دیجیتال را بر آن داشته است تا اندازه ساخت تراشه‌ها را به ابعاد کمتر از چندین نانومتر کاهش دهند. این پیشرفت‌ها باعث افزایش نمایی تعداد ترانزیستورها، افزایش فرکانس کاری مدار و کاهش ولتاژ کاری مدار شده‌اند؛ اما این کاهش ابعاد و مقیاس‌بندی ترانزیستورها، موجب ایجاد مشکلات جدیدی برای طراحی و ساخت تراشه‌های الکترونیکی شده است. ازجمله‌ی مهم‌ترین این مشکلات، افزایش آسیب‌پذیری مدارهای دیجیتال در برابر منابع اشکال‌های محیطی و درنتیجه، کاهش قابلیت اطمینان مدارهای دیجیتال در مقیاس نانو است.

برخورد ذرات پرانرژی به سطح تراشه که موجب ایجاد یک پالس گذرا[1]در خروجی دروازه‌های منطقی می‌شوند، ازجمله منابعی می‌باشند که می‌توانند عملکرد صحیح مدارهای دیجیتال را تحت تاثیر قرار دهند[۱][۲]. این اشکال‌های گذرا[2]می‌توانند در طول مدار منتشرشده و مقدار ذخیره‌شده در یک سلول حافظه را تغییر دهند. به این خطا اصطلاحا *خطای نرم*[3]گفته می‌شود و معیاری را که میزان آسیب‌پذیری مدارها در برابر خطای نرم را نشان می‌دهد، *نرخ خطای نرم*[4](SER) می‌نامند[۳].

در نسل‌های گذشته تکنولوژی، وجود برخی شرایط باعث پوشش[5]بخش قابل‌توجهی از پالس‌های گذرا می‌گردید و از وقوع خطای نرم در مدار جلوگیری می‌نمود[۴][۵]. پوشش‌ها در مدارها به سه دسته پوشش منطقی، الکتریکی و زمانی دسته‌بندی می‌شوند:

[1] Transient pulse
[2] Transient Faults
[3] Soft error
[4] Soft Error Rate
[5] Masking

تاریخ دریافت : ۱۳۹۵/۰۲/۳۱ تاریخ پذیرش : ۱۳۹۵/۰۹/۲۸

- هنگامی‌که پالس گذرا در ورودی دروازه با توجه به مقادیر کنترلی دیگر ورودی‌ها، به خروجی اصلی منتقل نشود، ***پوشش منطقی*** رخ داده است.

- با توجه به خصوصیات الکتریکی دروازه‌ها، عبور پالس گذرا از دروازه‌ها باعث تضعیف و حتی در مواردی از بین رفتن پالس گذرا می‌شود که اصطلاحا ***پوشش الکتریکی*** رخ داده است.

- اگر پالس گذرا در ورودی فلیپ فلاپ در بازه زمانی موسوم به پنجره نمونه‌برداری[6] حضور نداشته باشد، اصطلاحا ***پوشش زمانی*** رخ داده است.

پوشش‌های منطقی، الکتریکی و زمانی عوامل تاثیرگذاری در انتشار پالس گذرا در مدارهای ترکیبی می‌باشند [۶][۷][۸]. یکی از راه‌های کاهش نرخ خطای نرم مدارهای ترکیبی افزایش پوشش منطقی می‌باشد. پوشش منطقی زمانی رخ می‌دهد که پالس گذرای حاضر در یکی از ورودی‌های دروازه منطقی به دلیل مقدار منطقی در ورودی‌های دیگر آن دروازه تحت تاثیر قرار گرفته و خروجی بدون توجه به پالس گذرا بر اساس ورودی (یا ورودی-های دیگر) دروازه منطقی تعیین می‌شوند. به‌عبارت‌دیگر، برای اینکه یک پالس گذرا منجر به خطا شود، بایستی پالس گذرا از محل تولید به یکی از فلیپ فلاپ‌ها منتشر شود؛ در این صورت از محل تولید تا یکی از فلیپ فلاپ‌های مدار یک مسیر *تحریک‌شده*[7] وجود دارد. بنابراین، یکی از راه‌های کاهش نرخ خطای نرم، کاهش تعداد مسیرهای تحریک‌شده می‌باشد و به‌این‌ترتیب، احتمال این که یک پالس گذرای تولید شده در یکی از دروازه‌ها، مسیری به یکی از خروجی‌ها بیابد، کاهش می‌یابد. میزان پوشش منطقی که در یک مدار منطقی رخ می‌دهد متأثر از ساختار و توپولوژی‌ای است که تابع مورد نیاز مدار را پیاده-سازی می‌کند. بنابراین، برای استفاده از این پوشش به منظور کاهش نرخ خطای نرم بایستی ساختار مدار تغییر پیدا کند. البته این تغییر نباید تابع منطقی مدار را تغییر دهد، بلکه بایستی تنها دروازه‌ها و ارتباط بین آن‌ها طوری تغییر کند که نرخ خطای نرم در آن مدار کاهش یابد.

در این مقاله، یک روش جدید به‌منظور افزایش تحمل‌پذیری مدارهای ترکیبی در برابر خطای نرم مبتنی بر جایگزینی منطقی محلی ارائه می‌شود. در روش ارائه شده، در هر مرحله، زیرمداری[8] از مدار اصلی انتخاب می‌شود و سپس، به کمک روش QM توسعه‌یافته، پیاده‌سازی‌های ممکن از زیرمدار به دست می‌آید و بهترین پیاده‌سازی از زیرمدار از لحاظ افزایش پوشش منطقی انتخاب‌شده و در مدار اصلی جایگزین می‌شود. به‌منظور انتخاب بهترین جایگزین از بین مجموعه پیاده‌سازی‌های مختلف زیرمدار، معیار جدیدی به نام احتمال خرابی سراسری[9] (GFP) ارائه شده است. این معیار، با در نظر گرفتن احتمال ایجاد پالس-های گذرای خطادار در زیرمدار و احتمال انتشار این پالس‌ها توسط زیرمدار، این امکان را فراهم می‌آورد که با تغییر محلی ساختار زیرمدار بتوان به ارزیابی سراسری زیرمدار در ایجاد و انتشار پالس‌های گذرا پرداخت. به عبارت دیگر، در هر مرحله بدون نیاز به جایگزین نمودن پیاده‌سازی‌های مختلف به‌دست‌آمده از زیرمدار در مدار اصلی و محاسبه نرخ خطای نرم کل مدار به ازای هرکدام از آن‌ها، تاثیر هر یک را در نرخ خطای نرم مدار اصلی در نظر می‌گیرد. نتایج حاصل از شبیه‌سازی‌ها بر روی مدارهای محک ISCAS'85 نشان می‌دهد که روش پیشنهادی به‌طور میانگین موجب ۱۷/۷۵ ٪ کاهش نرخ خطای نرم به ازای ۵/۳۹ ٪ سربار مساحت در مقایسه با مدار اصلی شده است.

ادامه این مقاله بدین‌صورت سازمان‌یافته است. در ابتدا، به‌مرور روش‌های کاهش نرخ خطای نرم می‌پردازد. سپس، انگیزش روش پیشنهادی در این مقاله ارائه می‌شود. در گام بعدی، روش پیشنهادی به‌منظور کاهش نرخ خطای نرم مبتنی بر جایگزینی منطقی محلی را شرح می‌دهیم. سپس، نتایج شبیه‌سازی ارائه می‌شود و درنهایت، به نتیجه‌گیری پرداخته خواهد شد.

کارهای پیشین

روش‌های مقاوم‌سازی مدارها در برابر خطای نرم به دو دسته کلی تقسیم می‌شوند: ۱) دسته‌ای از روش‌ها که احتمال تولید اشکال گذرا (امکان این‌که اشکالی در خروجی دروازه منطقی به وجود بیاید) را کاهش می‌دهند و ۲) دسته‌ای از روش‌ها که احتمال انتشار اشکال گذرا (احتمال این‌که اشکال تولیدشده، در مدار منتشرشده و توسط عناصر حافظه ذخیره گردد) را کاهش می‌دهند. دسته دوم، از اثر پوشش‌های سه‌گانه (منطقی، الکتریکی و زمانی) برای کاهش نرخ خطای نرم استفاده می‌کنند.

بخشی از این روش‌ها تلاش دارند با تغییر خصوصیات الکتریکی دروازه‌های مدار، پوشش الکتریکی را افزایش داده و از ذخیره-شدن پالس‌های گذرا در عناصر حافظه جلوگیری نمایند. در [۹][۶]، از روش اندازه‌گذاری دروازه‌ها به‌منظور کاهش نرخ خطای نرم استفاده می‌کنند. بدین‌صورت که با تغییر نسبت عرض به طول ترانزیستورها در دروازه‌هایی که بیشترین سهم را در بروز خطای نرم در مدار دارند، مقاوم می‌شوند. در [۱۰]، روشی ارائه

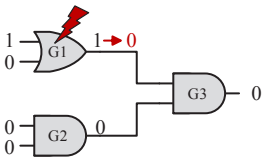
[6] Latching window
[7] Sensitized path
[8] Sub-circuit
[9] Global Failure Probaility

شده است که با در نظر گرفتن مقادیر مختلف ولتاژ تغذیه برای دروازه‌های مدار سعی در کاهش نرخ خطای نرم مدار دارد. بخشی از این روش‌ها تلاش دارند با تغییر خصوصیات زمانی مدارها، پوشش زمانی را افزایش داده و از ذخیره‌شدن پالس‌های گذرا در عناصر حافظه جلوگیری نمایند. در [۱۱][۱۲]، چارچوبی ارائه شده که از بین مجموعه موجود از فلیپ فلاپ‌ها، فلیپ فلاپی را انتخاب می‌کند که در عین حفظ کارایی و سرعت مدار، با طولانی کردن بازه‌های پنجره زمانی، احتمال پوشش زمانی را افزایش می‌دهد. در [۹]، ترکیبی از دو روش اندازه‌گذاری دروازه‌ها و انتخاب فلیپ‌فلاپ استفاده کرده تا به سطوح بالاتری از تحمل‌پذیری در برابر نرخ خطای نرم دست پیدا کند.

دسته دیگری از تحقیقات با تغییر خصوصیات منطقی مدارها، پوشش منطقی را افزایش داده و از ذخیره شدن پالس‌های گذرا در عناصر حافظه جلوگیری می‌نمایند. در [۱۳]، یک روش پوشش خطا مبتنی بر TMR[1] ارائه شده که تنها نودهایی که بیشترین تاثیر در نرخ خطای نرم مدار دارند، سه‌گانه می‌شوند. در [۱۴] نیز یک روش مبتنی بر TMR ارائه شده است به‌طوری‌که تنها خروجی‌های اصلی مدار که با احتمال بیشتری دچار خطا می‌شوند؛ محافظت می‌شوند. در [۱۵] روشی بر مبنای اضافه و حذف افزونگی[1] ارائه شده است. در این روش، با نگه‌داری سیم‌ها و دروازه‌هایی که بیشترین تاثیر را در پوشش دارند و حذف سیم‌ها و دروازه‌هایی که بیشترین تاثیر را در خطا دارند، سعی در کاهش نرخ خطای نرم مدار دارند. در [۱۶] نیز روشی ارائه شده که با اضافه نمودن سیم‌های اضافی به مدار، موجب کاهش خطای نرم می‌شود. در [۱۷] روشی مبتنی بر ترکیبی از اضافه و حذف افزونگی و اندازه‌گذاری ارائه شده است که با افزایش دو پوشش منطقی و الکتریکی پایداری مدارها را در برابر خطای نرم افزایش می‌دهد. در [۱۸] به منظور کاهش توان دینامیک[2] و همچنین کاهش خطای نرم مدارهای ترکیبی یک روش مبتنی بر هسته[3] ارائه شده است. در این روش با شناسایی بخش‌هایی از مدار که تحت شرایط خاصی از مقادیر ورودی تاثیری در مقدار خروجی ندارند، مدار اصلی به چندین زیر مدار بخش‌بندی[4] شده و برای مقادیر کنترلی مدار تنها یکی از زیر مدارها فعال می‌باشد و مقادیر خروجی را تعیین می‌کند و درنتیجه، گذارهای تک رخدادی[5] که در زیر مدار غیرفعال رخ می‌دهند به‌صورت منطقی پوشش داده‌شده و به خروجی انتشار پیدا نمی‌کند. در [۱۹]، چارچوبی مبتنی بر شبیه‌سازی ارائه شده که مدار اصلی به زیر-مدارهایی تک خروجی دسته‌بندی‌شده و سپس با پیکربندی مجدد زیرمدارها به‌طوری‌که منجر به افزایش پوشش منطقی شوند، موجب کاهش نرخ خطای نرم مدار می‌شود.

انگیزش: مشکل تغییر جزیی و ارزیابی سراسری

هنگامی‌که پالس گذرا در ورودی دروازه با توجه به مقادیر کنترلی دیگر ورودی‌ها، به خروجی اصلی منتقل نشود، پوشش منطقی رخ داده است. برای مثال، شکل ۱، یک مدار منطقی شامل سه دروازه را نشان می‌دهد. همان‌طور که مشاهده می‌شود، اگر خطایی در دروازه G_1 رخ دهد و مقدار خروجی دروازه از '۱'منطقی به '۰' منطقی تغییر کند، با توجه به ورودی دیگر دروازه G_3، مقدار خروجی مدار تغییر نمی‌کند. درنتیجه، پالس گذرا در این نقطه از مدار به‌طور منطقی پوشش داده می‌شود و به نقاط دیگر مدار منتشر نمی‌شود.

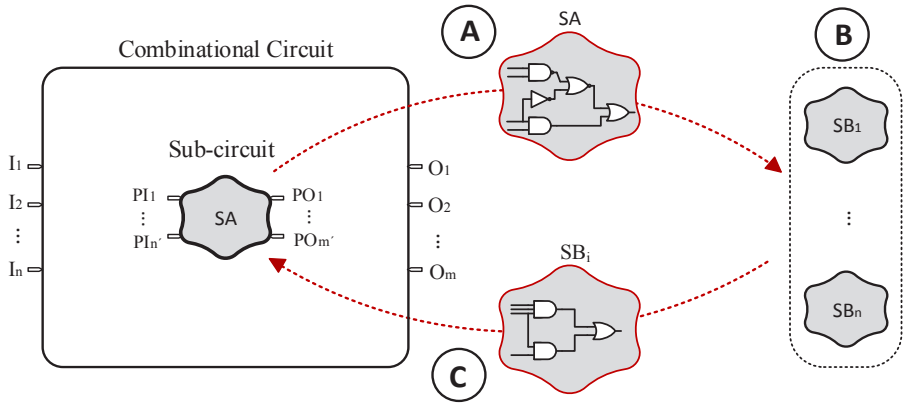


شکل ۱. پالس خطا ایجادشده در دروازه G1، به‌صورت منطقی در دروازه G3 پوشش داده می‌شود.

همان‌طور که بیان شد، برای استفاده از افزایش پوشش منطقی برای کاهش نرخ خطای نرم، بایستی ضمن حفظ تابع منطقی مدار، دروازه‌ها و ارتباط بین آن‌ها طوری تغییر یابد که موجب کاهش نرخ خطای نرم در آن مدار شود. تغییر ساختار به‌منظور افزایش پوشش منطقی، در مدارهای بزرگ (با تعداد دروازه‌های زیاد) امری زمان‌بر و در مواردی غیرممکن خواهد بود. به‌منظور ارزیابی تحمل‌پذیری مدار در برابر خطای نرم، نیاز به پیمایش مدار به ازای تمام دروازه‌های مدار می‌باشد. محدودیت این فرآیند زمانی آشکار می‌شود که با تغییر جزی ساختار زیرمداری از مدار، به دلیل تغییر دروازه‌های آن و ارتباط بین آن‌ها، به‌منظور ارزیابی تحمل‌پذیری مدار نیاز به محاسبه مجدد نرخ خطای نرم مدار باشد. در شرایطی که به‌منظور افزایش تحمل‌پذیری مدار، به‌صورت مداوم نیاز به تغییر ساختار زیرمدارهایی از مدار می-باشد، حجم زیادی از محاسبات صرف تخمین نرخ خطای نرم کل مدار می‌شود. به‌عبارت‌دیگر، در هر تغییر زیرمدار نیاز به محاسبه مجدد نرخ خطای نرم مدار می‌باشد. در این مقاله، به‌منظور رفع محدودیت تغییر جزئی و ارزیابی سراسری و درنتیجه کاهش سربار محاسبات، یک معیار ارزیابی سراسری جدید ارائه شده است.

---

1 Triple Modular Redundancy (TMR)
1 Redundancy addition and removal (RAR)
1 Dynamic power 2
1 Kernel-Based 3
1 Partition 4
1 Single Event Transient (SET) 5

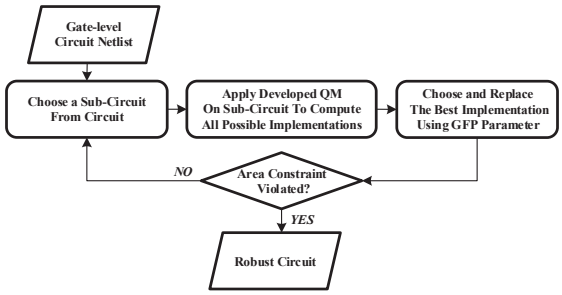


شکل ۲. فرآیند تغییر ساختار و جایگزینی محلی یک زیرمدار در مدار اصلی

شکل ۲، فرآیند جایگزینی محلی و تغییر ساختار یک زیرمدار در مدار اصلی با n ورودی اصلی ($I_1, I_2, \ldots, I_n$) و m خروجی اصلی ($O_1, O_2, \ldots, O_n$) را نشان می‌دهد. همان‌طور که مشاهده می‌شود، این فرآیند شامل سه مرحله می‌باشد. در مرحله A، زیرمدار SA از مدار اصلی انتخاب می‌شود. در مرحله B، پیاده‌سازی‌های مختلف ($SB_1, SB_2, \ldots, SB_n$) از زیرمدار به دست می‌آید. در مرحله C، بهترین پیاده‌سازی (SB_i) مبتنی بر پارامتر GFP انتخاب‌شده و جایگزین زیرمدار SA در مدار اصلی می‌شود.

روش پیشنهادی به‌منظور کاهش نرخ خطای نرم مبتنی بر جایگزینی منطقی محلی

شکل ۳، فلوچارت روش پیشنهادی جهت کاهش نرخ خطای نرم مدارهای ترکیبی را نشان می‌دهد. همان‌طور که مشاهده می‌شود، ابتدا، زیرمداری از مدار اصلی انتخاب می‌شود. سپس، به کمک روش QM توسعه‌یافته پیاده‌سازی‌های ممکن از زیرمدار به دست می‌آید. در ادامه، بهترین پیاده‌سازی از زیرمدار بر مبنای پارامتر GFP انتخاب‌شده و در مدار اصلی جایگزین می‌شود. این فرآیند تا زمانی که محدودیت مساحت برقرار باشد، ادامه می‌یابد. در ادامه، جزئیات روش پیشنهادی را بیان می‌نماییم.

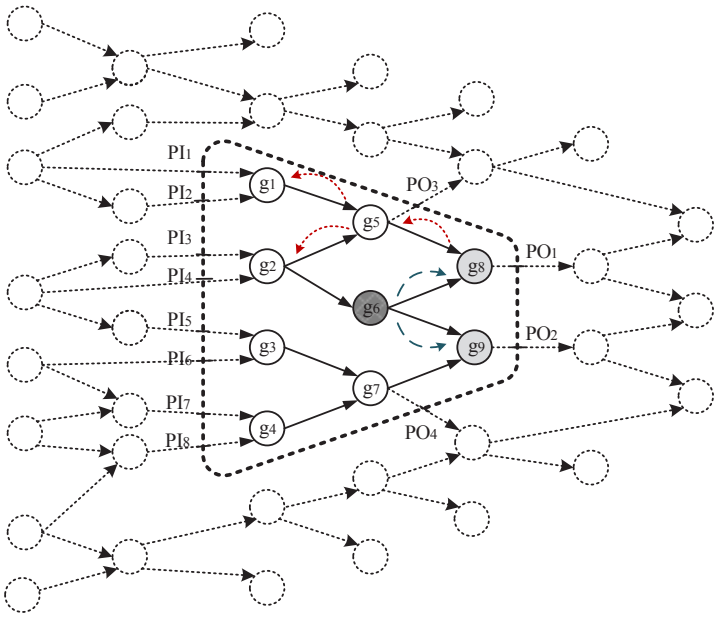


شکل ۳. فلوچارت روش پیشنهادی به‌منظور بهبود نرخ خطای نرم مدارهای ترکیبی مبتنی بر جایگزینی منطقی محلی

استخراج زیرمدار

به‌منظور باز-سنتز منطقی محلی، زیرمدارهایی از مدارهای اصلی استخراج می‌شوند. هر زیرمدار می‌تواند چندین ورودی (n') و چندین خروجی (m') داشته باشد. از طرفی، اندازه زیرمدار نقش مهمی در کیفیت و زمان موردنیاز جهت باز-سنتز دارد. در حالتی که اندازه زیرمدار بزرگ باشد، امکان به دست آوردن پیاده‌سازی‌های متفاوت از آن بیشتر بوده و درنتیجه امکان به دست آوردن پیاده‌سازی که موجب کاهش مسیرهای سنتز شده و افزایش پوشش منطقی شود، افزایش می‌یابد؛ ولی زمان لازم برای به دست آوردن پیاده‌سازی‌های متفاوت افزایش می‌یابد. از طرف دیگر، در حالتی که زیرمدار کوچک باشد، امکان به دست آوردن پیاده‌سازی‌های متفاوت از آن کمتر بوده و درنتیجه، امکان به دست آوردن پیاده‌سازی که موجب کاهش مسیرهای سنتز شده و افزایش پوشش منطقی شود، کاهش می‌یابد اما، زمان به دست آوردن پیاده‌سازی‌های متفاوت کاهش می‌یابد. به‌منظور سنتز مجدد زیرمدار، نیاز به محاسبه تمام[۱] PI ها (شامل اساسی) می‌باشد. ازآنجایی‌که تعداد PI وابسته به تعداد ورودی‌ها می‌باشد، تعداد ورودی‌های زیرمدار (n') را به مقدار ۱۰ محدود کرده‌ایم. بنابراین، می‌توان در مدت‌زمان کوتاهی تمام PI ها را محاسبه کرد. استخراج زیرمدار مبتنی بر انتخاب دروازه‌ای از زیرمدار می‌باشد. بدین‌صورت که با انتخاب دروازه‌ای از زیرمدار، دروازه‌های متصل به خروجی آن به‌عنوان خروجی زیرمدار در نظر گرفته می‌شوند و سپس از دروازه‌های خروجی زیرمدار به‌اندازه L سطح مدار اصلی به سمت ورودی‌های اصلی پیمایش می‌شود. در این حین، اگر دروازه‌ای در زیرمدار وجود داشته باشد که خروجی آن به دروازه‌ای غیر از دروازه‌های زیرمدار متصل باشد به‌عنوان خروجی زیرمدار در نظر گرفته می‌شود. شکل ۴، الگوریتم استخراج زیرمدار را نشان می‌دهد.

[1] prime implicant



شکل ۵. نحوه استخراج زیرمدار

Algorithm 1 : Sub-Circuit extraction

1. **input** = combinational circuit: ***C*** ;
   max input of Sub-Circuit: ***n´*** ;
   level: ***L*** ;
2. **output** = Sub-Circuit: ***SC*** ;
   outputs list of Sub-Circuit: L_{output} ;
3. **begin**
4. choose gate g_r from C ;
5. add fanout gates of g_r to L_{output} ;
6. copy L_{output} to fanout list L_{fo} ;
7. **For each**(g_i in L_{output})
8. statring from g_i, traverse C in reverse order for L level, and add gates to SC under the following conditions:
9. - prune branching when reaching L^{th} level ;
10. - terminate adding gates to SC when reaching n' ;
11. **end for each** ;
12. **For each**(g_i in L_{fo})
13. **if**(all fanout gates of g_i does not exist in SC)
14. add g_i to L_{output} ;
15. **end if** ;
16. **end foreach** ;
17. **end**

شکل ۴. الگوریتم استخراج زیرمدار

شکل ۵، مثالی از نحوه استخراج زیرمدار را نشان می‌دهد. در این مثال، تعداد ورودی زیرمدار برابر با n´=10 و تعداد سطح برابر با L=2 در نظر گرفته شده است. در این مثال، دروازه g6 انتخاب‌شده است. پس‌ازآن، دروازه‌های موجود در مخروط خروجی آن به‌عنوان خروجی زیر مدار در نظر گرفته می‌شود (دروازه‌های g8 و g9). سپس، مدار اصلی از دروازه‌های خروجی زیرمدار به‌اندازه دو سطح به سمت ورودی‌های اصلی پیمایش می‌شود. دراین‌بین، دروازه‌هایی وجود دارند که خروجی آن‌ها به دروازه‌هایی غیر از دروازه‌های زیرمدار متصل شده‌اند (دروازه‌های g5 و g7) که درنتیجه، آن‌ها را به‌عنوان خروجی‌های زیرمدار در نظر گرفته می‌شود.

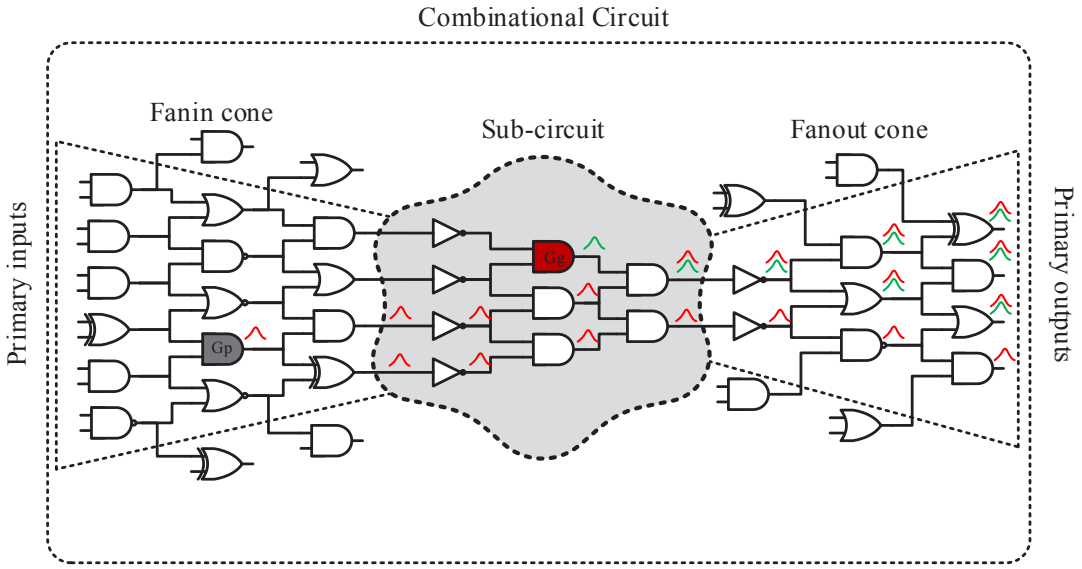
معیار ارزیابی محلی زیر مدار

برای هر زیرمدار، مخروط ورودی و مخروط خروجی را به‌صورت زیر تعریف می‌نماییم:

- *مخروط ورودی* زیرمدار برابر با دروازه‌هایی از مدار اصلی است که مسیری از آن‌ها به ورودی‌های زیرمدار وجود داشته باشد.
- *مخروط خروجی* زیرمدار برابر با دروازه‌هایی از مدار اصلی که مسیری از خروجی زیرمدار به آن‌ها وجود داشته باشد.

شکل ۶، مثالی از مخروط‌های ورودی و خروجی زیرمدار در مدار اصلی را نشان می‌دهد. با توجه به موقعیت زیرمدار در مدار اصلی، اندازه‌ی مخروط ورودی و خروجی زیرمدار متفاوت خواهد بود.

با در نظر گرفتن مخروط ورودی و مخروط خروجی، می‌توان نقش یک زیرمدار در نرخ خطای نرم مدار ترکیبی را به **ایجاد پالس-های خطا** و **انتشار پالس‌های خطا** دسته‌بندی نمود. نقش هر زیرمدار در ایجاد پالس‌های خطا هنگامی است که پالس‌های خطا در دروازه‌های موجود در زیرمدار تولید شوند و



شکل ۶. مخروط ورودی و مخروط خروجی زیرمدار و تاثیر ایجاد پالس خطا در زیرمدار و تاثیر انتشار پالس خطا از طریق زیرمدار

این پالس‌ها از طریق مخروط خروجی زیرمدار منتشرشده و در فلیپ فلاپ متصل به خروجی‌های اصلی مدار ذخیره شوند. نقش هر زیرمدار در انتشار پالس‌های خطا نیز مربوط به حالتی می‌باشد که پالس‌های خطا در دروازه‌های موجود در مخروط ورودی زیرمدار ایجاد شوند و از طریق این زیرمدار منتشرشده و در فلیپ فلاپ متصل به خروجی‌های اصلی مدار ذخیره شوند.

برای مثال، در شکل ۶، تاثیر ایجاد پالس خطا در زیرمدار نشان داده شده است. همان‌طور که مشاهده می‌شود، پالس خطا در دروازه‌ای (G_g) از زیرمدار ایجادشده (پالس سبزرنگ) و پس از انتشار به خروجی‌های زیرمدار، از طریق مخروط خروجی زیرمدار به خروجی‌های اصلی مدار می‌رسد. همچنین، این شکل، تاثیر انتشار پالس خطا توسط زیرمدار را نیز نشان می‌دهد. همان‌طور که مشاهده می‌شود، پالس خطا در دروازه‌ای از مخروط ورودی زیرمدار (G_p) ایجادشده (پالس قرمزرنگ) و پس از انتشار آن توسط زیرمدار، از طریق مخروط خروجی زیرمدار به خروجی‌های اصلی مدار می‌رسد.

از آنجایی یک زیرمدار می‌تواند هم در ایجاد و هم در انتشار پالس‌های خطا تاثیرگذار باشد، به‌منظور ارزیابی سراسری زیرمدار، معیار Global Failure Probability (GFP) که هر دو تاثیر را شامل می‌گردد را برای زیر مدار SC_i به‌صورت زیر معرفی می‌نماییم:

$$GFP_{SC_i} = P_{generation} + P_{propagation} \qquad (۱)$$

در این رابطه، $P_{generation}$ برابر با احتمال ایجاد پالس‌های خطا در زیرمدار و تبدیل آن به خطای نرم و همچنین، $P_{propagation}$ برابر با احتمال انتشار پالس‌های خطای ورودی توسط زیرمدار و تبدیل آن به خطای نرم می‌باشد.

احتمال ایجاد پالس‌های خطا در زیرمدار ($P_{generation}$) برابر با احتمال ایجاد پالس خطا به ازای تمام دروازه‌های زیرمدار و ذخیره شدن آن در فلیپ فلاپ متصل به خروجی‌های اصلی مدار می‌باشد. بنابراین، احتمال ایجاد پالس خطا در زیرمدار برابر است با :

$$P_{generation} = \sum_{i=1}^{N(PG)} LEG(G_i) \qquad (۲)$$

در این رابطه، $LEG(G_i)$ برابر با احتمال ایجاد پالس خطا به ازای دروازه G_i در زیرمدار و ذخیره شدن آن در فلیپ فلاپ متصل به خروجی‌های اصلی مدار و $N(PG)$ برابر با تعداد دروازه‌های تشکیل‌دهنده زیرمدار می‌باشد.

با توجه به شکل ۶، مسیر تبدیل پالس خطا ایجادشده در دروازه G_i در زیرمدار به خطای نرم شامل دو مرحله زیر می‌باشد:

۱) ایجاد پالس خطا در دروازه‌ای از زیرمدار و انتشار آن تا خروجی‌های زیرمدار؛
۲) انتشار پالس خطا از خروجی‌های زیرمدار به خروجی‌های اصلی مدار.

مقادیر منتشرشده در مرحله (۱) وابسته به ساختار زیرمدار می‌باشد. همچنین، مقادیر منتشرشده در مرحله (۲) وابسته به ساختار مخروط خروجی زیرمدار می‌باشد. ازآنجایی‌که تغییر ساختار زیرمدار تنها در مقادیر منتشرشده در مرحله (۱) را تغییر می‌دهد، می‌توان مراحل (۱) و (۲) را از یکدیگر جدا در نظر گرفت.

بنابراین، احتمال ایجاد پالس خطا به ازای دروازه G_i در زیرمدار و ذخیره شدن آن در فلیپ فلاپ متصل به خروجی‌های اصلی مدار به‌صورت زیر محاسبه می‌شود:

$$LEG(G_i) = \sum_{k=1}^{N(PO)} EPP_{G_i}(PO_K) \times SERG(PO_K) \qquad (۳)$$

در این رابطه، N(PO) برابر با تعداد خروجی‌های زیر مدار می‌باشد. $EPP_{Gi}(PO_k)$ احتمال ذخیره‌شدن پالس خطا منتشرشده از دروازه G_i در فلیپ‌فلاپی که فرض می‌شود به خروجی PO_k زیرمدار متصل شده است؛ می‌باشد (مرحله ۱). $SERG(PO_k)$ برابر است با احتمال ایجاد پالس خطا در خروجی k ام زیرمدار (PO_k) و انتشار و ذخیره آن در یکی از فلیپ فلاپ‌های متصل به خروجی‌های اصلی مدار می‌باشد (مرحله ۲).

انتشار پالس خطا از محل رخداد خطا تا خروجی‌های زیرمدار، تابعی از ساختار زیرمدار می‌باشد. به‌منظور محاسبه احتمال انتشار پالس خطا از محل رخداد خطا تا خروجی‌های زیرمدار، از محل رخداد خطا (دروازه G_i) به سمت خروجی‌های زیرمدار، قوانین انتشار پالس خطا ارائه شده در [20] را تا خروجی‌های زیرمدار سطح به سطح به دروازه‌های زیرمدار اعمال می‌نماییم (شکل ۷ قسمت B). در انتها، احتمال ذخیره شدن پالس خطا در فلیپ‌فلاپ متصل به خروجی PO_k زیرمدار ($EPP_{Gi}(PO_k)$) با درنظر گرفتن پوشش منطقی، به‌صورت زیر محاسبه می‌شود:

$$EPP_{G_i}(PO_k) = P_{a_{G_i}}(PO_k) + P_{\bar{a}_{G_i}}(PO_k) \qquad (۴)$$

در این رابطه، $Pa_{Gi}(PO_K)$ احتمال آن است که سیگنال خروجی PO_K دارای مقدار خطا با قطبیت یکسان با مقدار خطا در محل برخورد در دروازه Gi می‌باشد. همچنین $P\hat{a}_{Gi}(PO_K)$ برابر با احتمال اینکه سیگنال خروجی PO_K دارای مقدار خطا با قطبیت غیر یکسان با مقدار خطا در محل برخورد در دروازه Gi می‌باشد.

انتشار پالس خطا از خروجی‌های زیرمدار به خروجی‌های اصلی مدار، از طریق مخروط خروجی زیرمدار صورت می‌گیرد. مخروط خروجی زیرمدار برابر با اجتماع تمام مخروط‌های خروجی مربوط به خروجی‌های زیرمدار می‌باشد. شکل ۷ قسمت C، مخروط خروجیِ مربوط به خروجی PO_1 زیرمدار را نشان می‌دهد. به دلیل اینکه مخروط‌های خروجی مربوط به خروجی‌های زیرمدار ساختارهای متفاوتی از یکدیگر دارند، درنتیجه نقش متفاوتی در انتشار پالس خطا از یکدیگر نشان می‌دهند. به‌منظور در نظر گرفتن نقش هر خروجی زیرمدار در انتشار پالس‌های خطا به خروجی‌های اصلی مدار و تبدیل آن‌ها به خطای نرم، به هر خروجی یک مقدار $SERG(PO_k)$ اختصاص می‌یابد. این مقدار، احتمال ایجاد پالس خطا در خروجی k ام زیرمدار (PO_k) و انتشار و ذخیره آن در فلیپ فلاپ متصل به خروجی‌های اصلی مدار را نشان می‌دهد:

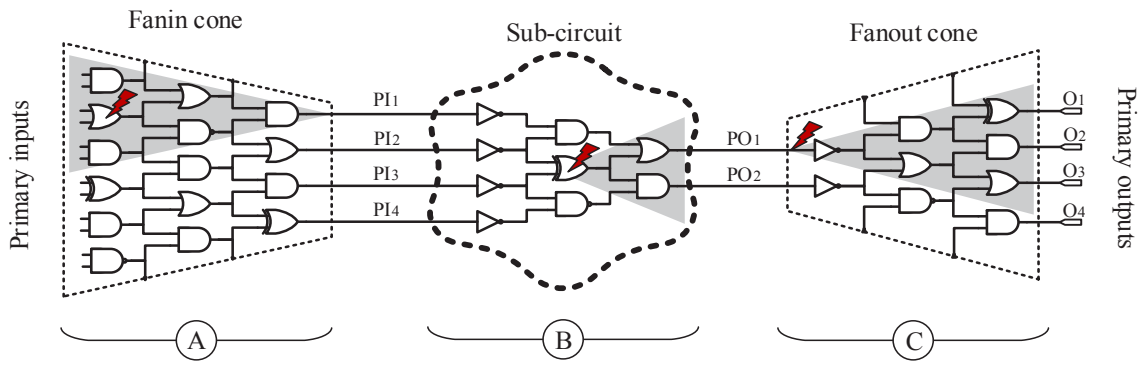
$$SERG(PO_k) = \sum_{j=1}^{NFO(PO_k)} EPP_j \qquad (۵)$$

در این رابطه، $NFO(PO_k)$ برابر با تعداد خروجی‌های اصلی مدار موجود در مخروط خروجیِ مربوط به خروجی k ام زیرمدار می باشد. همچنین، EPP_j برابر با احتمال ذخیره شدن پالس خطا در فلیپ‌فلاپ متصل به خروجی اصلی j ام موجود در مخروط خروجی مربوط به خروجی k ام زیرمدار (PO_k) می‌باشد.

احتمال انتشار پالس خطا از طریق زیرمدار ($P_{propagation}$) برابر با احتمال ایجاد پالس خطا به ازای تمام دروازه‌های موجود در مخروط ورودی زیرمدار، انتشار از طریق زیرمدار و ذخیره‌شدن آن در فلیپ فلاپ متصل به خروجی‌های اصلی مدار می‌باشد. بنابراین، احتمال انتشار پالس خطا از طریق زیرمدار برابر است با:

$$P_{propagation} = \sum_{j=1}^{N(PI)} LEP(PI_j) \qquad (۶)$$

در این رابطه، $LEP(PI_j)$ احتمال ایجاد پالس خطا در مخروط ورودیِ مربوط به ورودی PI_j زیرمدار و ذخیره شدن آن در فلیپ فلاپ متصل به خروجی‌های اصلی مدار و N(PI) برابر با تعداد ورودی‌های زیرمدار می‌باشد.



شکل ۷. (A) مخروط جلویی ورودی PI_1 زیرمدار و نحوه محاسبه $SERP(PI_1)$. (B) خطا در دروازه G_i زیرمدار و نحوه محاسبه $EPP_{Gi}(PO_k)$. (C) مخروط خروجی PO_1 زیرمدار و نحوه محاسبه $SERG(PO_1)$

روند تبدیل پالس خطا ایجادشده در مخروط ورودیِ مربوط به ورودی PI_j زیرمدار به خطای نرم شامل سه مرحله زیر می‌باشد:

۱) ایجاد پالس خطا در دروازه‌ای از مخروط ورودی زیرمدار و انتشار آن تا ورودی‌های زیرمدار
۲) انتشار پالس خطا از ورودی‌های زیرمدار تا خروجی‌های زیرمدار
۳) انتشار پالس خطا از خروجی‌های زیرمدار تا خروجی‌های اصلی مدار

مقادیر منتشرشده در مرحله (۱) وابسته به ساختار مخروط ورودی زیرمدار می‌باشد. مقادیر منتشرشده در مرحله (۲) وابسته به ساختار زیرمدار می‌باشد و مقادیر منتشرشده در مرحله (۳) وابسته به ساختار مخروط خروجی زیرمدار می‌باشد. ازآنجایی‌که تغیر ساختار زیرمدار تنها در مقادیر منتشرشده در مرحله (۲) را تغییر می‌دهد، می‌توان مراحل (۱)، (۲) و (۳) را مستقل از یکدیگر در نظر گرفت.

بنابراین، احتمال ایجاد پالس خطا در مخروط ورودیِ مربوط به ورودی (PI_j) زیرمدار و ذخیره شدن آن در فلیپ فلاپ متصل به خروجی‌های اصلی مدار به‌صورت زیر محاسبه می‌شود.

$$LEP(PI_j) = SERP(PI_j) \times LEG(PI_j) \qquad (۷)$$

در این رابطه، $SERP(PI_j)$ برابر است با احتمال ذخیره‌شدن پالس خطا منتشرشده از دروازه موجود در مخروط ورودیِ مربوط به ورودی PI_j در فلیپ‌فلاپی که فرض می‌شود به ورودی PI_jزیرمدار متصل شده است (مرحله ۱). $LEG(PI_j)$ برابر با احتمال ایجاد پالس خطا به ازای ورودی PI_j زیرمدار و ذخیره شدن آن در فلیپ فلاپ متصل به خروجی‌های اصلی مدار می‌باشد (مرحله ۲ و ۳).

انتشار پالس خطا از محل رخداد تا ورودی‌های زیرمدار، از طریق مخروط ورودی زیرمدار صورت می‌گیرد. مخروط ورودی زیرمدار برابر با اجتماع تمام مخروط‌های ورودی مربوط به ورودی‌های زیرمدار می‌باشد. شکل ۷ قسمت A، مخروط ورودیِ مربوط به ورودی PI_1 زیرمدار را نشان می‌دهد. به دلیل اینکه مخروط‌های ورودی مربوط به ورودی‌های زیرمدار ساختارهای متفاوتی با یکدیگر دارند، درنتیجه نقش متفاوتی در انتشار پالس خطا به ورودی زیرمدار با یکدیگر نشان می‌دهند. به‌منظور در نظر گرفتن نقش هر ورودی زیر مدار در انتشار پالس‌های خطا، به هر ورودی یک مقدار $SERP(PI_j)$ اختصاص می‌یابد. این مقدار، احتمال ایجاد پالس خطا در تمام دروازه‌های موجود در مخروط ورودیِ مربوط به ورودی j زیرمدار (PI_j) و انتشار آن‌ها به ورودی j زیر-مدار را نشان می‌دهد و به‌صورت زیر محاسبه می‌شود:

$$SERP(PI_j) = \sum_{h=1}^{NFI(PI_j)} EPP_{PI_j}(G_h) \qquad (۸)$$

که در آن، $NFI(PI_j)$ برابر با تعداد دروازه‌های موجود در مخروط ورودیِ مربوط به ورودی j ام زیرمدار می‌باشد. $EPP_{PIj}(G_h)$ برابر با احتمال ذخیره شدن پالس خطا ایجادشده در دروازه G_h در فلیپ فلاپی که فرض می‌شود به ورودی j زیرمدار متصل شده است؛ می‌باشد.

با توجه به روابط ۱ تا ۸، مقدار GFP_{SCi} برابر است با:

$$GFP_{SC_i} = \left[\sum_{i=1}^{N(PG)} \left(\sum_{k=1}^{N(PO)} EPP_{G_i}(PO_K) \times SERG(PO_K) \right)\right] \qquad (۹)$$
$$+ \left[\sum_{j=1}^{N(PI)} \left(SERP(PI_j) \times \left(\sum_{k=1}^{N(PO)} EPP_{PI_j}(PO_K) \times SERG(PO_K) \right)\right)\right]$$

همان‌طور که در رابطه ۹ مشاهده می‌شود، معیار GFP_{SCi} وابسته به مقادیر $SERP(PI_j)$، $SERG(PO_k)$ و $EPP(PO_k)$ می‌باشد. همان‌طور که قبلا هم گفته شد، $SERP(PI_j)$، تاثیر مخروط ورودی زیرمدار در انتشار پالس خطا و $SERG(PO_k)$، تاثیر مخروط خروجی زیرمدار در انتشار پالس خطا را در نظر می‌گیرد. همچنین، $EPP(PO_k)$ تاثیر ایجاد پالس‌های خطا از دروازه‌های زیرمدار و یا انتشار پالس‌های خطا از طریق ورودی‌های زیرمدار تا خروجی‌های زیرمدار را در نظر می‌گیرد. بنابراین، در ارزیابی هر زیرمدار و پیاده‌سازی‌های مختلف آن، با داشتن مقادیر $SERP(PI_j)$ مربوط به ورودی‌های زیرمدار و همچنین مقادیر $SERG(PO_k)$ مربوط به خروجی‌های زیرمدار تنها نیاز به محاسبه مجدد $EPP(PO_k)$ مربوط به زیرمدار می‌باشد. درنتیجه، با استفاده از معیار GFP_{SCi}، بدون نیاز به محاسبه مجدد نرخ خطای نرم کل مدار می‌توان با تغییر جزئی و محاسبات محلی به ارزیابی سراسری آن پرداخت.

سنتز مجدد زیرمدار با استفاده از روش QM توسعه‌یافته

به‌منظور افزایش پوشش منطقی مدار، نیاز به تغییر ساختار زیرمدار و به دست آوردن پیاده‌سازی‌های مختلف از آن می‌باشد. هرکدام از این پیاده‌سازی‌ها با توجه به ساختارشان، رفتار متفاوتی از یکدیگر در ایجاد و انتشار پالس‌ها خطا از خود نشان می‌دهند. درنتیجه ممکن است پوشش منطقی متفاوتی از یکدیگر نشان می‌دهند. برای به دست آوردن پیاده‌سازی‌های مختلف زیرمدار، روش QM را توسعه داده و استفاده می‌نماییم. در این روش، برخلاف روش QM، پیاده‌سازی‌های مختلف از زیرمدار بدون در نظر گرفتن هزینه طراحی به دست می‌آیند.

Algorithm 2 : Developed QM

1. **input** = Sub-Circuit: ***SC*** ;
2. **output** =all possible implementation of *SC*: ***APISC***;
3. **foreach**(output O_i of *SC*)
4. compute *ON-set* minterm of O_i , ON_{Oi} ;
5. add ON_{Oi} to M_{list} ;
6. **end foreach** ;
7. compute all of the *prime implicants* using *QM* method, *P* ;
8. make prime implicant table,T_{PI}, using *P* ;
9. using T_{PI} to compute all possible *PI sets* that cover M_{list} , S_{PIj} , and add S_{PIj} to API_{SC} ;
10. **return** API_{SC} ;

شکل ۸. الگوریتم سنتز مجدد زیرمدار با استفاده از روش QM توسعه‌یافته

Algorithm 3 : Optimization Algorithm

1. **input** = golden circuit: C_G
2. **output** = robust circuit: C_R
3. compute SER of golden circuit, $SER_{CG;}$
4. compute Area of golden circuit, $Area_{CG;}$
5. // compute SERG and SERP for all gates of circuit;
6. choose sub-circuit SC_i from circuit; // using *algorithm 1*
7. //extract SERG for all outputs of SC_i;
8. //extract SERP for all inputs of SC_i;
9. compute GFP parameter for sub-circuit SC_i, GFP_{SCi};
10. compute all possible implementations of SC_i using developed QM, API_{SCi}; // using *algorithm 2*
11. compute GFP for all implementations in API_{SCi};
12. Choose implementation I_m in API_{SCi} with least GFP_{Im};
13. **if**($GFP_{Im} < GFP_{SCi}$)
14. **swap** sub-circuit SC_i and implementation I_m;
15. **end If;**
16. **if**(Area constraint violated)
17. **goto** line 22;
18. **else**
19. **goto** line 7;
20. **end If;**
21. compute SER of robust circuit SER_{CR};
22. compute Area of robust circuit $Area_{CR}$;

شکل ۹. الگوریتم بهینه‌سازی به‌منظور کاهش نرخ خطای نرم مدار ترکیبی

شکل ۸، الگوریتم سنتز مجدد زیرمدار با استفاده از روش QM توسعه‌یافته را نشان می‌دهد. همان‌طور که مشاهده می‌شود، در ابتدا مجموعه مینترم‌های تمام خروجی‌های مدار (M_{list}) محاسبه می‌شود. پس‌ازآن، با استفاده از روش QM مجموعه PI ها (P) به دست می‌آیند. سپس، بدون در نظر گرفتن هزینه و با استفاده از مجموعه P، تمام مجموعه PI هایی که مجموعه M_{list} را پوشش دهند، به دست می‌آید. این روند تا جایی که تمام مینترم‌های مدار پوشش داده نشده‌اند، ادامه می‌یابد. درنهایت، مجموعه پیاده‌سازی‌های مختلف از یک مدار (API_{SC}) به دست می‌آید.

الگوریتم بهینه‌سازی

الگوریتم پیشنهادی به‌منظور کاهش نرخ خطای نرم مدارهای ترکیبی مبتنی بر جایگزینی منطقی محلی در شکل ۹ نشان داده شده است. همان‌طور که مشاهده می‌شود، این الگوریتم یک مدار ترکیبی را به‌عنوان ورودی می‌گیرد و نرخ خطای نرم و مساحت مدار اصلی محاسبه می‌شود. هم‌زمان با محاسبه نرخ خطای نرم مدار، مقادیر SERG و SERP برای تمام دروازه‌های مدار محاسبه می‌شوند. در گام بعدی، زیرمداری از مدار اصلی (SC_i) انتخاب می‌شود و پارامتر GFP_{SCi} متناظر با آن محاسبه می‌شود. سپس، به کمک روش QM توسعه‌یافته، مجموعه پیاده‌سازی‌های ممکن از زیر مدار (API_{SCi}) به دست می‌آید. در ادامه، با استفاده از معیار GFP، بهترین پیاده‌سازی زیرمدار (I_m) به‌طوری‌که رابطه ($GFP_{Im} < GFP_{SCi}$) برقرار باشد انتخاب‌شده و در مدار اصلی جایگزین زیرمدار می‌شود. این فرآیند تا زمانی که محدودیت مساحت برقرار باشد، ادامه می‌یابد.

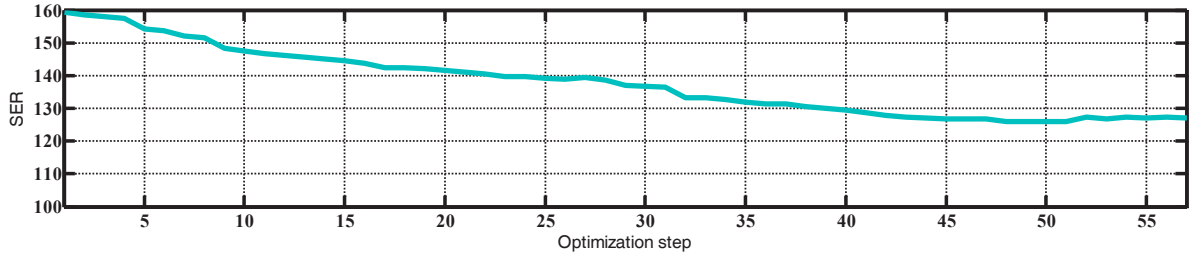
نتایج شبیه‌سازی

در این بخش نتایج شبیه‌سازی حاصل از روش پیشنهادی را بررسی می‌نماییم. روش پیشنهادی به زبان C++ پیاده‌سازی شده است. شبیه‌سازی‌ها بر روی مدارهای محک ISCAS'85 و MCNC صورت گرفته است. برای زیرمدارهای استخراج‌شده تعداد ورودی‌ها برابر با n´=10 و تعداد سطوح پیمایش شده برابر با L=3 در نظر گرفته شده است. همچنین، در این مقاله تنها محدودیت مساحت لحاظ شده است.

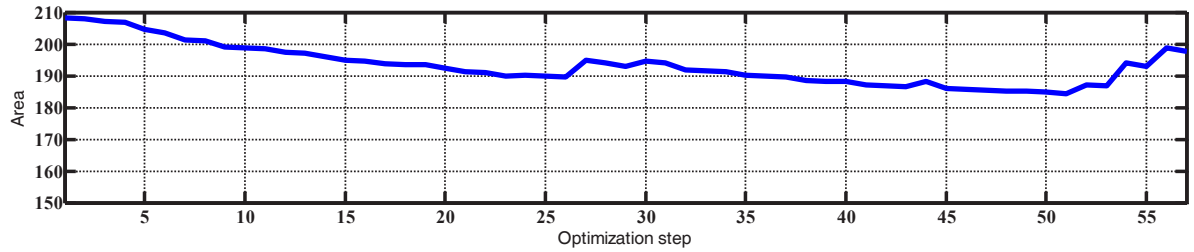
به‌منظور اطمینان از کاهش نرخ خطای نرم با استفاده از معیار ارائه‌شده‌ی GFP در هر مرحله از فرآیند بهینه‌سازی، شبیه‌سازی‌هایی برای دو مدار محک C432 و C2670 صورت گرفته است. میزان نرخ خطای نرم و سربار مساحت در هر مرحله از فرآیند بهینه‌سازی محاسبه شده است. شکل‌های ۱۰ و ۱۱ نمودار روند تغییر نرخ خطای نرم (قسمت الف) و سربار مساحت (قسمت ب) با استفاده از روش پیشنهادی را به ترتیب برای مدارهای محک C432 و C2670 نشان می‌دهند. همان‌طور که مشاهده می‌شود، روند تغییر نرخ خطای نرم برای هر دو مدار محک در اکثر مواقع

نزولی می‌باشد. بااین‌حال، در برخی مراحل نرخ خطای نرم با سرعت بیشتری کاهش‌یافته است. برای مثال، مراحل ۵، ۸ و۳۲ در مدار C432 (شکل ۱۰ قسمت الف) و مراحل اولیه در مدار C2670 (شکل ۱۱ قسمت الف) بهبود نرخ خطای نرم با سرعت بیشتری کاهش یافته است.

میزان کاهش نرخ خطای نرم در هر مدار وابسته به ساختار مدار اصلی و همچنین ساختار، اندازه و موقعیت زیرمدار استخراج‌شده در هر مرحله از فرآیند بهبود نرخ خطای نرم می‌باشد. این نتایج نشان می‌دهند که روش پیشنهادی بدون نیاز به محاسبه مجدد نرخ خطای نرم کل مدار در اکثر مراحل از فرآیند بهبود نرخ خطای نرم موثر می‌باشد. شکل ۱۰ قسمت ب و شکل ۱۱ قسمت ب نمودار سربار مساحت (Area) نیز نشان می‌دهد. همان‌طور که مشاهده می‌شود مساحت مدار در برخی موارد افزایش و در برخی موارد کاهش یافته است. همچنین، روند بهبود در مدار C432 در ۵۷ مرحله و در مدار C2670 در ۲۷۵ مرحله صورت گرفته است. از این مراحل به بعد، انتخاب زیرمدار و تغییر ساختار آن به‌منظور افزایش پوشش منطقی تاثیری قابل‌توجهی در بهبود نرخ خطای نرم نداشته است.

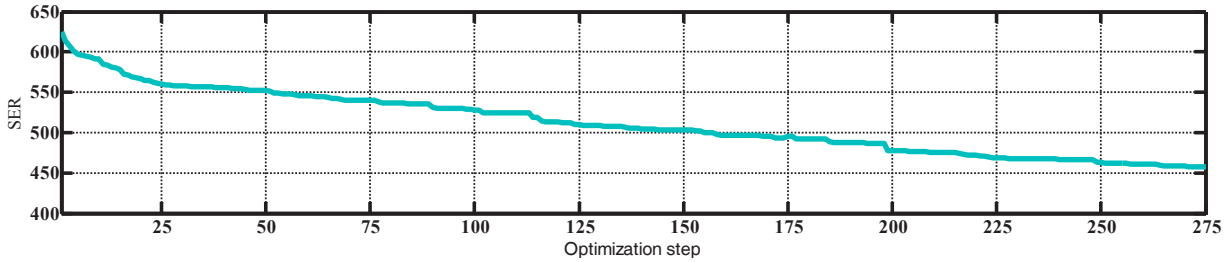


(الف)

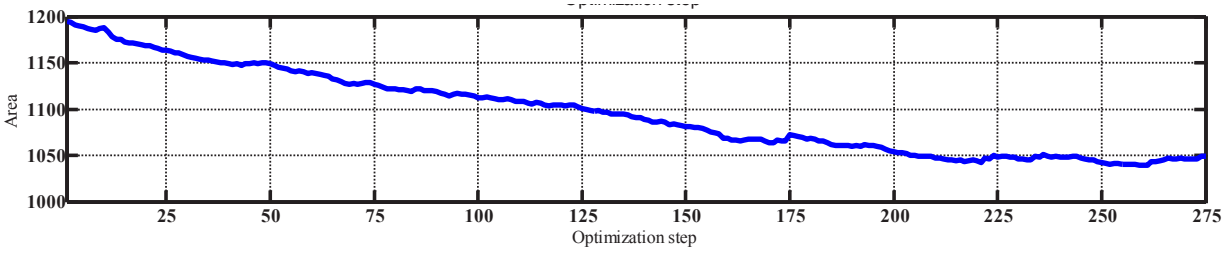


(ب)

شکل ۱۰. روند تغییر نرخ خطای نرم و سربار مساحت برای مدار محک C432



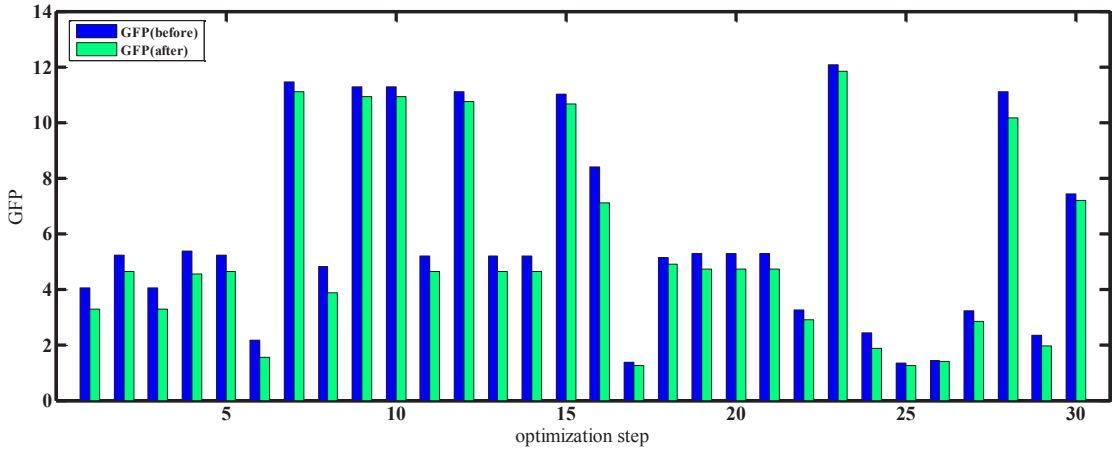
(الف)



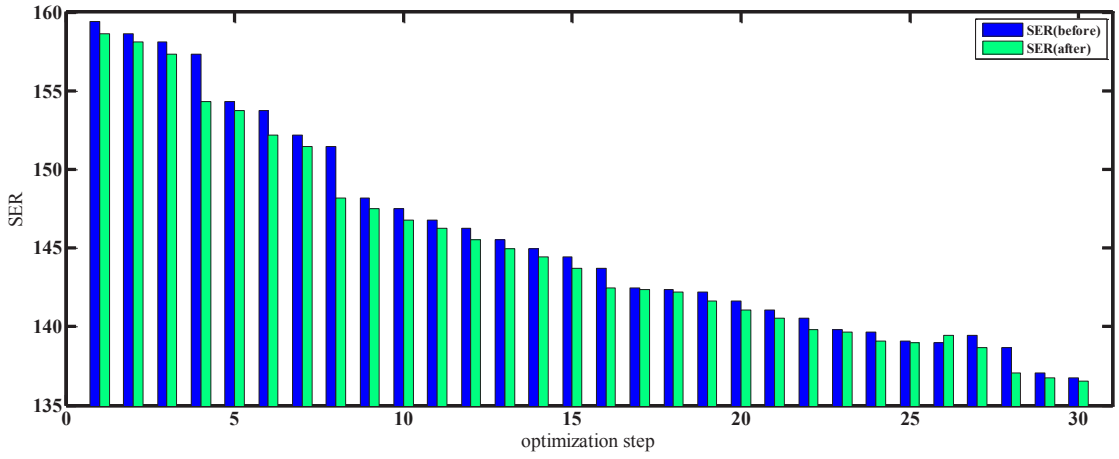
(ب)

شکل ۱۱. روند تغییر نرخ خطای نرم و سربار مساحت برای مدار محک C2670

همچنین، به‌منظور اطمینان از کاهش نرخ خطای نرم با استفاده از معیار ارزیابی سراسری زیرمدار (GFP)، در هر مرحله مقادیر GFP زیرمدار، قبل و بعد از فرآیند بهینه‌سازی مربوط به مدارهای محک C432 و C2670 اندازه‌گیری شده است. شکل‌های ۱۲ و ۱۳ مقادیر معیار GFP (قسمت الف) و نرخ خطای نرم (قسمت ب) در ۳۰ مرحله نخست بهینه‌سازی را به ترتیب برای مدارهای محک C432 و C2670 نشان می‌دهد. همان‌طور که مشاهده می‌شود، در هر مرحله از فرآیند، مقدار GFP کاهش‌یافته (شکل ۱۲ و ۱۳ قسمت الف) و درنتیجه موجب کاهش نرخ خطای نرم مدار (شکل ۱۲ و ۱۳ قسمت ب) می‌شود. به‌عبارت‌دیگر، با تغییر ساختار زیرمدار در هر مرحله به‌طوری‌که موجب کاهش GFP شود، نرخ خطای نرم مدار کاهش می‌یابد.

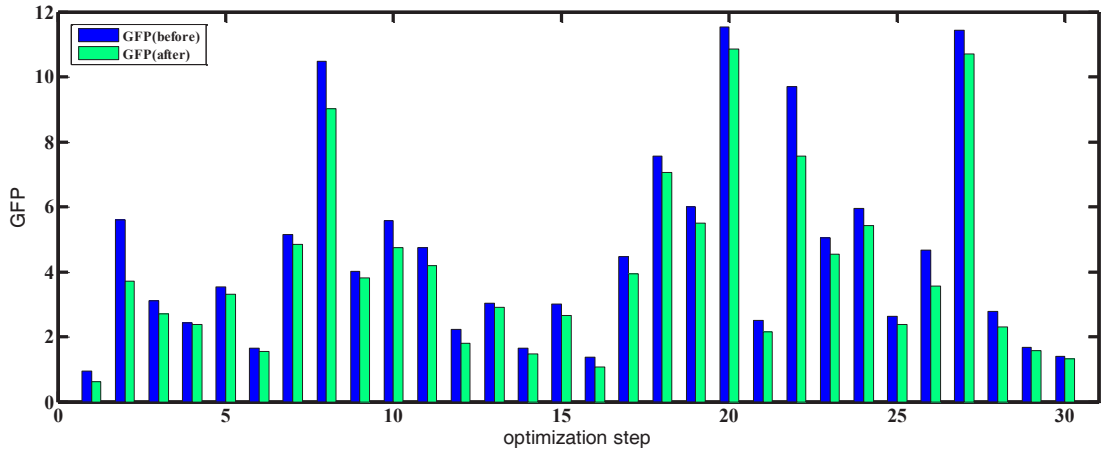


(الف)

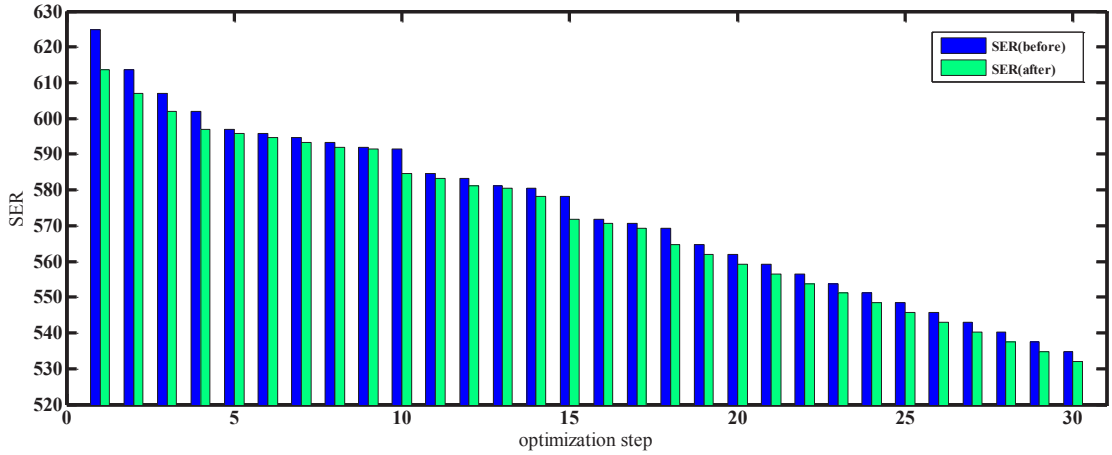


(ب)

شکل ۱۲. معیار ارزیابی محلی زیرمدار (GFP) و نرخ خطای نرم در ۳۰ مرحله نخست مدار محک C432



(الف)



(ب)

شکل ۱۳. معیار ارزیابی محلی زیرمدار (GFP) و نرخ خطای نرم در ۳۰ مرحله نخست مدار محک C2670

جدول ۱، نتایج حاصل از شبیه‌سازی بر روی مدارهای محک ISCAS'85 و MCNC را نشان می‌دهد. در ستون اول اطلاعاتی از تعداد ورودی‌ها (Input)، خروجی‌ها (Output) و دروازه‌های (Gate) مدارهای محک ارائه‌شده است. سپس، نرخ خطای نرم (SER) و مساحت (Area) کل مدار در قبل و بعد از فرآیند بهینه-سازی نشان داده‌شده است. درنهایت، میزان کاهش نرخ خطای نرم (SER Reduction) و سربار مساحت (Area Overhead) ارائه‌شده است. بیشترین کاهش نرخ خطای نرم مربوط به مدار C6288 و کمترین آن مربوط به مدار C880 می‌باشد. همان‌طور که قبلا هم به آن اشاره شد، میزان کاهش نرخ خطای نرم در هر مدار وابسته به ساختار مدار اصلی و همچنین ساختار، اندازه و موقعیت زیرمدار استخراج‌شده در هر مرحله از فرآیند بهبود نرخ خطای نرم می‌باشد. همان‌طور که مشاهده می‌شود، نتایج آزمایش‌ها نشان می‌دهد که روش پیشنهادی به‌طور میانگین موجب ۱۷/۷۵% کاهش نرخ خطای نرم به ازای ۵/۳۹% سربار مساحت در مقایسه با مدار اصلی متناظر شده است.

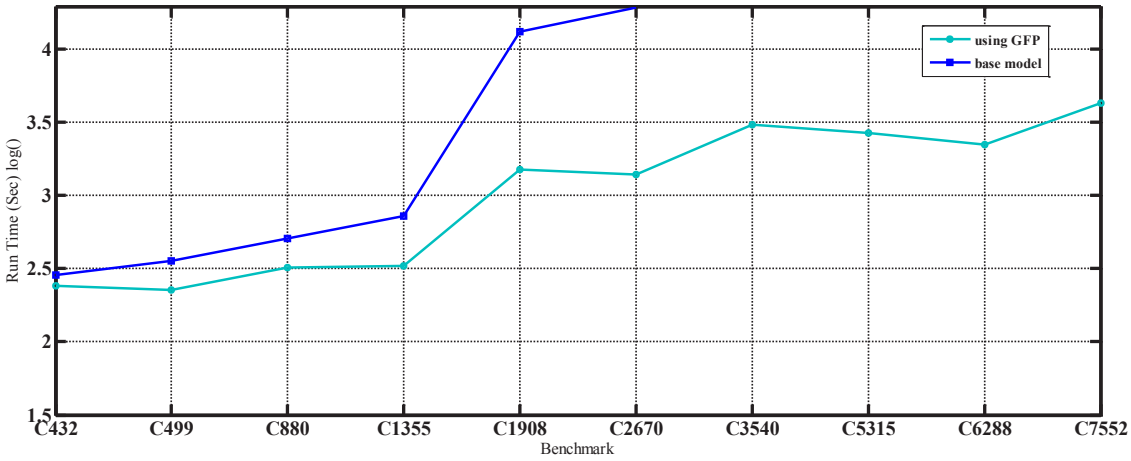
همچنین، روش پیشنهادی با روش ارائه شده در مرجع [۱۹] را مورد مقایسه قرار می‌دهیم. در این مقاله روش [۱۹] را پیاده سازی نموده‌ایم. این روش، چارچوبی مبتنی بر شبیه‌سازی می-باشد که مدار اصلی به زیرمدارهایی تک خروجی دسته‌بندی‌شده و سپس با پیکربندی مجدد زیرمدارها به طوری که منجر به افزایش پوشش منطقی شوند، موجب کاهش نرخ خطای نرم مدار می‌شود. جدول ۲، نتایج حاصل از مقایسه‌ی روش پیشنهادی و روش مشابه ارائه شده در مرجع [۱۹]، نشان می‌دهد. همان‌طور که مشاهده می‌شود، ستون اول میزان نرخ خطای نرم (SER)، ستون دوم سربار مساحت (Area change) و ستون سوم زمان

جدول ۱. نتایج شبیه‌سازی بر روی مدارهای محک

اطلاعات مدار				قبل از اعمال بهینه‌سازی		بعد از اعمال بهینه‌سازی		مقایسه	
Bench	Input	Output	Gate	SER	Area	SER	Area	SER Reduction (%)	Area Overhead (%)
C432	36	7	233	159.40	208.28	127.15	197.64	20.23	-5.11
C499	41	32	638	146.89	434.11	129.62	461.59	11.76	6.33
C880	60	26	433	164.69	372.40	150.36	388.08	8.7	4.21
C1355	41	33	629	185.33	487.31	166.70	526.15	10.05	7.97
C1908	33	25	425	286.14	807.31	232.72	884.00	18.67	9.5
C2670	157	64	872	624.88	1194.87	457.60	1048.84	26.77	-12.22
C3540	50	22	901	577.45	1648.40	493.60	1764.12	14.52	7.02
C5315	178	123	1833	970.89	2640.32	838.46	2544.74	13.64	-3.62
C6288	32	32	2788	4926.91	1987.55	3473.96	2234.20	29.49	12.41
C7552	207	108	2171	1160.75	3400.01	1010.08	3690.37	12.98	8.54
apex3	54	50	280	182.90	8134.60	148.95	8820.35	18.56	8.43
apex4	9	19	438	190.50	10443.18	167.45	11837.34	12.10	13.35
ex1010	10	10	810	724.29	16342.03	548.65	18147.82	24.25	11.05
spla	16	46	2296	5866.80	1960.06	4294.50	2111.38	26.80	7.72
Average	--	--	--	--	--	--	--	17.75	5.39

جدول ۲. مقایسه میزان نرخ خطای نرم، سربار مساحت و زمان بهینه‌سازی روش پیشنهادی و روش ارائه شده در مرجع [۱۹].

	SER (%)		Area change (%)		Run time (T_{opt}) (sec)		Comparison (روش پیشنهادی / روش [۱۹])	
Bench	روش پیشنهادی	روش [۱۹]	روش پیشنهادی	روش [۱۹]	روش پیشنهادی	روش [۱۹]	SER	Run time
C432	11.50	17.80	-5.11	45.90	241.20	4180.00	1.55	17.33
C499	18.80	26.90	6.33	39.90	226.80	7540.00	1.43	33.25
C880	25.66	34.50	4.21	41.90	322.20	5960.00	1.34	18.50
C1355	16.59	18.80	7.97	38.30	333.00	13200.00	1.13	39.64
C1908	19.40	23.00	9.50	37.50	1513.80	26000.00	1.19	17.18
C2670	24.17	37.20	-12.22	46.60	1393.20	66100.00	1.54	47.44
C3540	17.30	-	7.02	-	3075.00	-	-	-
C5315	27.22	-	-3.62	-	2680.20	-	-	-
C6288	23.15	-	12.41	-	2239.80	-	-	-
C7552	21.45	-	8.54	-	4279.80	-	-	-
apex3	22.40	27.10	8.43	31.30	325.70	8370.49	1.21	25.70
apex4	21.90	24.60	13.35	46.50	480.90	9425.64	1.12	19.60
ex1010	23.50	28.80	11.05	43.70	1525.20	40417.80	1.23	26.50
spla	17.70	-	7.72	-	2466.80	-	-	-
Average	20.77	26.52	5.40	41.29	1507.40	20132.66	1.30	27.24



شکل ۱۴. زمان بهینه‌سازی (T_{opt}) نرخ خطای نرم مدارهای محک ISCAS'85

بهینه‌سازی (T_{opt}) ارائه شده است. همان‌طور که ملاحظه می گردد، روش پیشنهادی در حدود ۲۷ برابر سریع‌تر از روش [۱۹] می‌باشد. از آن‌جایی‌که روش [۱۹] مبتنی بر شبیه‌سازی می‌باشد، بهینه‌سازی مدارها در زمان زیادی صورت می‌گیرد و حتی در مواردی که تعداد گیتهای مدار زیاد باشد (مانند C7552،C6288،C5315،C3540) امکان‌پذیر نخواهد بود. همان‌طور که نتایج نشان می‌دهد، روش پیشنهادی نرخ خطای نرم کمتری در مقایسه با روش [۱۹] دارد. روش ارائه شده در [۱۹]، موجب نرخ خطای نرم به میزان ۲۶/۵۲ ٪ به ازای ۴۱/۲۹ ٪ سربار مساحت شده است. از طرفی، روش پیشنهادی موجب نرخ خطای نرم به میزان ۲۰/۷۷ ٪ به ازای ۵/۴۰ ٪ سربار مساحت شده است. لازم به ذکر است که نحوه تخمین و مقادیر نرخ خطای نرم نشان داده شده در این جدول مبتنی‌بر شبیه‌سازی می‌باشد. بدین صورت که تعداد مشخصی از اشکال‌های گذرا به ازای ورودی‌های متفاوت مدار، به مدار اعمال شده و تعداد خطاهای منتشر شده به خروجی مدار محاسبه می‌شود.

همچنین شکل ۱۴، تفاوت زمان اجرای بهینه‌سازی (Topt) نرخ خطای نرم مدارهای محک با استفاده از روش پایه نسبت به روش پیشنهادی را نشان می‌دهد (توجه شود که محور y بر حسب

لگاریتم می‌باشد). در روش پایه، در هر زیرمدار برای انتخاب بهترین پیاده‌سازی متناظر نیاز به جایگزین نمودن تمام پیاده-سازی‌ها و محاسبه مجدد نرخ خطای نرم به ازای هر پیاده‌سازی می‌باشد. در روش پیشنهادی، در هر زیرمدار برای انتخاب بهترین پیاده‌سازی متناظر از معیار GFP استفاده می‌شود. همان‌گونه که مشاهده می‌شود با استفاده از روش پیشنهادی، زمان بهینه‌سازی به‌طور میانگین ۸/۶ برابر کاهش یافته است. نکته قابل توجه آن است که استفاده از روش پایه بر خلاف روش پیشنهادی برای مدارهای محک بزرگتر از C3540 چندین روز به طول می‌انجامد و امکان‌پذیر نخواهد بود.

نتیجه‌گیری

در این مقاله، یک ایده جدید برای افزایش تحمل‌پذیری مدارهای ترکیبی در مقابل خطای نرم معرفی‌شده است. ایده پیشنهادی بر اساس تکنیک باز-سنتز منطقی مبتنی بر جایگزینی منطقی محلی به‌منظور کاهش نرخ خطای نرم مدارهای ترکیبی می‌باشد. در این روش، به کمک توسعه‌ای از روش QM پیاده‌سازی‌های مختلف از یک زیرمدار مستخرج از مدار اصلی به‌دست‌آمده و سپس، با استفاده از معیار پیشنهادی GFP، از بین پیاده‌سازی-های مختلف زیرمدار بهترین جایگزین باهدف کاهش نرخ خطای نرم تعیین می‌شود. مزیت اصلی این معیار این است که در هر مرحله از فرآیند بهبود نرخ خطای نرم، بدون نیاز به جایگزین نمودن تمام پیاده‌سازی‌های مختلف از زیرمدار در مدار اصلی و محاسبه مجدد نرخ خطای نرم کل مدار به ازای هرکدام از آن‌ها، می‌توان تاثیر هر یک را در نرخ خطای نرم مدار اصلی محاسبه نمود. نتایج حاصل از شبیه‌سازی‌ها بر روی مدارهای محک ISCAS'85 نشان می‌دهد که روش پیشنهادی به‌طور میانگین موجب ٪۱۷/۷۵ کاهش نرخ خطای نرم به ازای ٪۵/۳۹ سربار مساحت در مقایسه با مدار اصلی شده است.

روش پیشنهادی در این مقاله تنها پارامتر پوشش منطقی را جهت کاهش نرخ خطای نرم در نظر می‌گیرد. توسعه الگوریتم معرفی‌شده، جهت در نظر گرفتن پوشش الکتریکی و زمانی یکی از کارهای آتی در راستای بهبود روش پیشنهادی می‌باشد.

منابع

[1] M. Nicolaidis, *Soft errors in modern electronic systems*, vol. 41. Springer Science & Business Media, 2010.

[2] R. C. Baumann, "Radiation-induced soft errors in advanced semiconductor technologies," *Device Mater. Reliab. IEEE Trans.*, vol. 5, no. 3, pp. 305–316, 2005.

[3] V. Ferlet-Cavrois, L. W. Massengill, and P. Gouker, "Single event transients in digital CMOS—A review," *Nucl. Sci. IEEE Trans.*, vol. 60, no. 3, pp. 1767–1790, 2013.

[4] H.-M. Chou, M.-Y. Hsiao, Y.-C. Chen, K.-H. Yang, J. Tsao, C.-L. Lung, S.-C. Chang, W.-B. Jone, and T.-F. Chen, "Soft-error-tolerant design methodology for balancing performance, power, and reliability," *Very Large Scale Integr. Syst. IEEE Trans.*, vol. 23, no. 9, pp. 1628–1639, 2015.

[5] D. A. Black, W. H. Robinson, I. Z. Wilcox, D. B. Limbrick, and J. D. Black, "Modeling of Single Event Transients With Dual Double-Exponential Current Sources: Implications for Logic Cell Characterization," *Nucl. Sci. IEEE Trans.*, vol. 62, no. 4, pp. 1540–1549, 2015.

[6] Q. Zhou and K. Mohanram, "Gate sizing to radiation harden combinational logic," *Comput. Des. Integr. Circuits Syst. IEEE Trans.*, vol. 25, no. 1, pp. 155–166, 2006.

[7] P. Shivakumar, M. Kistler, S. W. Keckler, D. Burger, and L. Alvisi, "Modeling the effect of technology trends on the soft error rate of combinational logic," in *Dependable Systems and Networks, 2002. DSN 2002. Proceedings. International Conference on*, 2002, pp. 389–398.

[8] S. Rezaei, S. Ghassem, H. Asadi, and M. Fazeli, "Microelectronics Reliability Soft error estimation and mitigation of digital circuits by characterizing input patterns of logic gates," *Microelectron. Reliab.*, 2014.

[9] R. R. Rao, D. Blaauw, and D. Sylvester, "Soft error reduction in combinational logic using gate resizing and flipflop selection," in *Computer-Aided Design, 2006. ICCAD'06. IEEE/ACM International Conference on*, 2006, pp. 502–509.

[10] K.-C. Wu and D. Marculescu, "Power-aware soft error hardening via selective voltage scaling," in *Computer Design, 2008. ICCD 2008. IEEE International Conference on*, 2008, pp. 301–306.

[11] P. V. Torvi, V. R. Devanathan, and V. Kamakoti, "Framework for selective flip-flop replacement for soft error mitigation," in *VLSI Design (VLSID), 2015 28th International Conference on*, 2015, pp. 381–386.

[12] P. V. Torvi, V. R. Devanathan, A. Vanjari, and V. Kamakoti, “SER mitigation technique through selective flip-flop replacement,” in *Quality Electronic Design (ASQED), 2015 6th Asia Symposium on*, 2015, pp. 25–30.

[13] K. Mohanram and N. A. Touba, “Partial error masking to reduce soft error failure rate in logic circuits,” in *Defect and Fault Tolerance in VLSI Systems, 2003. Proceedings. 18th IEEE International Symposium on*, 2003, pp. 433–440.

[14] A. H. El-maleh and F. Chikh, “Microelectronics Reliability A generalized modular redundancy scheme for enhancing fault tolerance of combinational circuits,” *Microelectron. Reliab.*, vol. 54, no. 1, pp. 316–326, 2014.

[15] K.-C. W. K.-C. Wu and D. Marculescu, “Soft error rate reduction using redundancy addition and removal,” in *Design Automation Conference, 2008. ASPDAC 2008. Asia and South Pacific*, 2008, pp. 559–564.

[16] S. Almukhaizim and Y. Makris, “Soft error mitigation through selective addition of functionally redundant wires,” *IEEE Trans. Reliab.*, vol. 57, no. 1, pp. 23–31, 2008.

[17] K.-C. Wu and D. Marculescu, “A low-cost, systematic methodology for soft error robustness of logic circuits,” *Very Large Scale Integr. Syst. IEEE Trans.*, vol. 21, no. 2, pp. 367–379, 2013.

[18] N. N. Mahatme, N. J. Gaspard, T. Assis, I. Chatterjee, T. D. Loveless, B. L. Bhuva, W. H. Robinson, L. W. Massengill, S.-J. Wen, and R. Wong, “Kernel-Based Circuit Partition Approach to Mitigate Combinational Logic Soft Errors,” *Nucl. Sci. IEEE Trans.*, vol. 61, no. 6, pp. 3274–3281, 2014.

[19] A. H. El-Maleh and K. A. K. Daud, “Simulation-Based Method for Synthesizing Soft Error Tolerant Combinational Circuits,” *Reliab. IEEE Trans.*, vol. 64, no. 3, pp. 935–948, 2015.

[20] G. Asadi and M. B. Tahoori, “An analytical approach for soft error rate estimation in digital circuits,” in *Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on*, 2005, pp. 2991–2994.