

## الگوریتم مسیریابی هوشمند تحمل‌پذیر اشکال در شبکه روی تراشه

فرزین قطبی وایقان<sup>\*</sup>، مصطفی ارسالی صالحی نسب<sup>\*\*</sup>

<sup>\*</sup> دانشکده‌ی مهندسی برق و کامپیوتر، دانشگاه آزاد اسلامی واحد قزوین، قزوین، ایران

<sup>\*\*</sup> پردیس دانشکده‌های فنی، دانشکده‌ی مهندسی برق و کامپیوتر، دانشگاه تهران، تهران، ایران

[f.ghotbi@qiau.ac.ir](mailto:f.ghotbi@qiau.ac.ir), [mersali@ut.ac.ir](mailto:mersali@ut.ac.ir)

### چکیده

رشد افزایش تعداد ترانزیستورها بر روی تراشه در سال‌های اخیر دچار مشکل شده است. هرچند، تکنولوژی VLSI امکان قراردادن میلیون‌ها ترانزیستور بر روی یک تراشه را بوجود آورده اما مسئله سیم‌بندی بین ترانزیستورها و فضای اشغالی آنها یک مشکل جدی است. پیدایش سیستم بر روی تراشه‌ها توانست این مشکل را تا حدودی برطرف نماید اما استفاده از گذرگاه نیز خود مشکلاتی از جمله گسترش ناپذیری را به دنبال داشت. برای حل مشکلات موجود در سیستم بر روی تراشه‌ها ایده‌ی استفاده از شبکه بر روی تراشه مطرح گردید. با توجه به گسترش استفاده از شبکه بر روی تراشه مسئله تحمل‌پذیری اشکال در شبکه روی تراشه اهمیت قابل توجهی پیدا می‌کند، یکی از راه‌های بالابردن قابلیت اطمینان در شبکه بر روی تراشه استفاده از الگوریتم‌های مسیریابی تحمل‌پذیر اشکال است. به همین منظور در این مقاله الگوریتمی پویا و هوشمند ارائه شده است که با توجه به وجود خطاهای دائمی و گذرا در سطح لینک و مسیریاب قادر است بسته‌ها را بدون از بین رفتن آنها در شبکه جابه‌جا کرده و به مقصد برساند. نتایج شبیه‌سازی حاکی از آن است که این الگوریتم نسبت به الگوریتم FTYX<sup>۱</sup> در حضور خرابی‌های دائمی و موقتی از نظر از دست دادن بسته، عملکرد قابل قبولی را دارا است.

### کلمات کلیدی

سیستم بر روی تراشه - شبکه بر روی تراشه - تحمل‌پذیری اشکال - الگوریتم‌های مسیریابی - خطای دائمی - خطای گذرا

گردید [۲]، [۴] و [۵]. جهت بالا بردن ضریب اطمینان کارکرد شبکه بر روی تراشه‌ها روش‌های مختلفی به کار گرفته شد، اما تقریباً در تمامی روش‌های انجام شده تعدادی از بسته‌ها در شبکه از بین می‌روند. در این مقاله الگوریتمی پویا و هوشمند جهت تحمل‌پذیری اشکال ارائه شده است که ۱۰۰٪ تضمین می‌نماید در صورت وجود مسیری از مبدا به مقصد، بسته را سالم به مقصد ارسال نماید و همچنین همانطور که در ادامه خواهید دید، الگوریتم پیشنهادی فاقد Livelock و بن‌بست می‌باشد. در ادامه در بخش ۲ ابتدا مروری خواهیم داشت بر کارهای گذشته، سپس در بخش ۳ الگوریتم مسیریابی پیشنهادی توضیح داده خواهد شد و در بخش ۴ محیط تست و شبیه‌سازی معرفی می‌شود و در بخش ۵ نیز نتایج حاصل از شبیه‌سازی آورده شده است و در بخش ۶ نتیجه‌گیری از این الگوریتم بیان شده است و نهایتاً در بخش آخر مراجع استفاده شده آمده است.

### ۲- بررسی کارهای گذشته

روش‌های مختلفی در زمینه تحمل‌پذیری اشکال در شبکه روی تراشه مطرح شده است که از آن جمله می‌توان به روش‌های سخت‌افزاری مانند استفاده از مسیرهای پشتیبان پیش‌فرض [۶] اشاره کرد که در

### ۱- مقدمه

رشد روز افزون تکنولوژی سیلیکون امکان قرار دادن میلیون‌ها ترانزیستور در یک تراشه را به وجود آورده است [۳]. اما یکی از مشکلات بسیار جدی مسئله سیم‌بندی بین این ترانزیستورها می‌باشد بدیهی است که سیم‌ها، فضای بسیار زیادی را اشغال می‌کنند و فرآیند مسیریابی آنها بسیار پیچیده می‌شود. پیدایش SoC<sup>۲</sup> تا حدود زیادی مسئله مربوط به سیم‌بندی و همچنین مسیریابی را حل نمود، زیرا در SoC ها انتقال داده از طریق گذرگاه انجام می‌گیرد. استفاده از گذرگاه باعث کاهش فضای اشغالی می‌شود اما مشکل بزرگ گذرگاه این است که در زمان انتقال داده فقط یک بلاک می‌تواند گذرگاه را در اختیار گرفته و از آن استفاده نماید، بنابراین نمی‌توان این سیستم را گسترش داد از طرفی با افزایش تعداد بلاک‌های متصل به گذرگاه اثرات مقاومتی و خازنی بیشتر خواهد شد [۱]. بنابراین SoC ها با مشکل توان مصرفی، مدیریت درخت کلاک و همچنین سرعت مواجه هستند. لازم به ذکر است که به دلیل استفاده از گذرگاه که خود گلوگاه سیستم است، نمی‌توان از منابع به درستی استفاده کرد. برای حل این مشکل در سال‌های اخیر ایده‌ی استفاده از شبکه بر روی تراشه مطرح