مقاله پژوهشی

شبکه بر تراشه با ولتاژ تطبیقی چندسطحی برای مدیریت حافظه توان‌آگاه در پردازنده‌های چندهسته‌ای

سیده معصومه مؤمنی و هادی‌شهریار شاه‌حسینی

چکیده: مقیاس‌بندی ولتاژ، یک روش پرکاربرد برای کاهش مصرف انرژی است که هزینه آن، افزایش تأخیر در شبکه در سامانه‌های بر تراشه چندپردازنده‌ای است. برای کاهش این هزینه عملکردی بر شبکه و سیستم، کاهش میزان جابه‌جایی داده‌ها و ارتباط‌ها در شبکه باید مورد توجه قرار گیرد. در برنامه‌های کاربردی حافظه‌محور و ارتباط‌محور، بخش قابل توجهی از تأخیر شبکه به دلیل ترافیک ناشی از عدم دسترسی به حافظه نهان است. در این مقاله از روش مقیاس‌بندی ولتاژ به صورت تطبیقی و چندسطحی استفاده می‌کنیم؛ در حالی که از فضای خالی حافظه میانگیرهای ورودی در گره‌های شبکه بر تراشه برای کاهش ترافیک ناشی از عدم دسترسی به حافظه نهان استفاده می‌شود. بنابراین روش پیشنهادی باعث افزایش کارایی حافظه و کاهش مصرف انرژی تراشه می‌شود. به‌منظور حصول بیشینه ظرفیت ناشی از به‌کارگیری رویکرد مقیاس‌بندی ولتاژ، ولتاژ منابع در سه سطح مختلف و با توجه به میزان متوسط فضای خالی حافظه میانگیرهای شبکه اعمال می‌شود. به این صورت که وقتی حافظه میانگیرها نزدیک به پر هستند، به‌کارگیری مقیاس‌بندی ولتاژ متوقف می‌شود. روش پیشنهادی به طور متوسط، میزان عدم دسترسی به داده در حافظه نهان را ۱۶ درصد و مصرف انرژی را ۱۲٫۵ درصد بهبود می‌دهد.

کلیدواژه: **چندپردازنده، سامانه بر تراشه، مدیریت انتقال داده‌ها، حافظه نهان، مدیریت انرژی.**

۱- مقدمه

شبکه بر تراشه (NoC) به‌عنوان بستر اصلی برای ارتباط داده‌ها در سطح تراشه، تأثیر قابل توجهی بر عملکرد کلی سامانه‌های چندپردازنده‌ای روی تراشه (MPSoC) دارد [۱] و [۲]. با پیشرفت فناوری و افزایش تعداد هسته‌ها در پردازنده‌های چندهسته‌ای که به دلیل پیچیدگی برنامه‌ها به وجود آمده‌اند، ترافیک و تأخیر روی تراشه افزایش می‌یابد که این امر منجر به افزایش مصرف انرژی می‌شود [۳]. در میان طرح‌های مدیریت انرژی، رویکرد مقیاس‌بندی ولتاژ که ولتاژ مسیریاب‌ها و پیوندها (لینک‌ها) را کاهش می‌دهد، می‌تواند در بهره‌وری انرژی NoC مؤثر باشد [۴]. با این حال، این رویکرد کمی دقت خروجی را کاهش می‌دهد و سبب جابه‌جایی اضافی داده‌ها برای ارسال مجدد بسته‌های خراب در شبکه می‌شود؛ بنابراین باعث افزایش تأخیر در شبکه می‌شود.

برای به‌کارگیری بهینه روش مقیاس‌بندی ولتاژ، باید تعداد بسته‌های ارسال‌شده در شبکه تا حد امکان کاهش یابد تا بر چالش مذکور غلبه شود. اساساً انتخاب روش مسیریابی مناسب یکی از رویکردهای دستیابی به این هدف است. یکی از روش‌ها استفاده از الگوریتم مسیریابی مناسب با میانگین تعداد گام کم جهت انتقال داده است. هرچه تعداد گام‌ها در مسیر انتخاب‌شده در شبکه کمتر باشد، استفاده کمتری از مسیریاب‌ها که عناصر پرمصرف در شبکه هستند، صورت می‌پذیرد. بنابراین یک روش مسیریابی مناسب می‌تواند مصرف انرژی را نیز کاهش دهد. رویکردهای مبتنی بر معماری و الگوریتم زیادی برای مدیریت حرارتی در زمان اجرا وجود دارد که تعداد بسته‌ها را در شبکه کاهش می‌دهد [۵]. روش ارتباطات تقریبی، یکی از روش‌هایی است که سعی می‌کند حجم داده‌های مبادله‌شده در شبکه را کاهش می‌دهد. کاهش زمان دسترسی به منابع توزیع‌شده واقع در سایر گره‌های شبکه، مانند حافظه نهان سطح ۲ (L۲)، روش دیگری برای کاهش ازدحام است. جانشانی مناسب وظایف و مدیریت داده‌ها دو راهبرد اصلی هستند که برای کاهش زمان دسترسی به داده‌ها استفاده شده‌اند [۲]. در این مقاله، روش مقیاس‌بندی ولتاژ به صورت تطبیقی پیشنهاد شده که مقیاس‌بندی ولتاژ، متناسب با ظرفیت آزاد منابع مورد استفاده برای ذخیره‌سازی داده‌ها که یک روش مدیریت داده روی تراشه است، اعمال می‌شود. هرچه مقدار فضای آزاد در منابع گره‌ها برای ذخیره داده‌ها بیشتر باشد، مقدار ولتاژ بیشتر کاهش می‌یابد.

در سامانه‌های بر تراشه چندپردازنده‌ای، معمولاً حافظه L۱، حافظه نهان محلی هر گره است و حافظه L۲، حافظه نهان مشترک است که بین تمام گره‌های شبکه توزیع شده است [۶]. واضح است که زمان دسترسی به حافظه نهان L۲ واقع در گره‌های دیگر (دسترسی غیرمحلی) بیشتر از زمان دسترسی به حافظه نهان L۲ محلی (دسترسی محلی) است. همچنین اگر بلوک متعلق به حافظه L۲ باشد، هر گونه فقدان داده در حافظه نهان در سطح ۱، باعث جابه‌جایی اضافی بسته‌ها برای جایگزینی بلوک خارج‌شده از L۱ می‌شود. هنگامی که برنامه کاربردی با فقدان داده در حافظه نهان مواجه می‌شود، بلوک خارج‌شده از L۱ به مسیریاب‌های محلی فرستاده می‌شود. در این مقاله برای کاهش ترافیک شبکه از فضای خالی حافظه میانگیر مسیریاب محلی برای ذخیره موقت بلوک‌های خارج‌شده از حافظه نهان L۱ استفاده شده و سازوکار جدیدی برای تعیین زمان ذخیره موقت پیشنهاد کرده‌ایم تا به این ترتیب به میزان بیشتری از انتقال بلوک‌های اخراج‌شده در شبکه پیشگیری شود. علاوه بر آن برای کاهش مصرف انرژی در شبکه، روش مقیاس‌بندی ولتاژ چندسطحی را به صورت تطبیقی و متناسب با فضای خالی حافظه میانگیر اعمال کرده‌ایم. مهم‌ترین ایده‌های ارائه‌شده در این مقاله به شرح زیر است:

- برای کاهش ترافیک در NoC، از سازوکار جدیدی به نام تحریک بر مبنای فضا برای تعیین زمان ذخیره موقت بلوک‌های خارج‌شده از

این مقاله در تاریخ ۱۳ تیر ماه ۱۴۰۳ دریافت و در تاریخ ۲۸ مهر ماه ۱۴۰۳ بازنگری شد.

سیده معصومه مؤمنی، دانشکده مهندسی برق، دانشگاه علم و صنعت ایران، تهران، ایران، (email: ma__momeni@elec.iust.ac.ir).

هادی‌شهریار شاه‌حسینی (نویسنده مسئول)، دانشکده مهندسی برق، دانشگاه علم و صنعت ایران، تهران، ایران، (email: hshsh@iust.ac.ir).

حافظه نهان L۱ در فضای خالی حافظه میانگیر مسیریاب‌های محلی استفاده می‌کنیم. به این ترتیب بدون سخت‌افزار اضافی، در صورت درخواست مجدد بلوک‌های خارج شده، به جای دسترسی غیرمحلی به حافظه نهان L۲، درخواست رابا فراخوانی بلوک از مسیریاب محلی پاسخ می‌دهیم..

- از روش مقیاس‌بندی ولتاژ به روش تطبیقی با سه سطح ولتاژ مختلف استفاده می‌کنیم؛ به طوری که سطح ولتاژ منابع شبکه با توجه به میانگین فضای خالی حافظه میانگیر ورودی مسیریاب‌ها تعیین می‌شود. بدین ترتیب می‌توان از بیشینه ظرفیت رویکرد مقیاس‌بندی ولتاژ بهره برد و به بهینه‌ترین میزان مصرف انرژی در شبکه و سیستم دست یافت.

در ادامه مقاله و در بخش ۲ پیشینه پژوهش بررسی می‌شود. در بخش ۳ روش پیشنهادی معرفی شده است. در بخش ۴ نتایج شبیه‌سازی‌ها و ارزیابی‌ها ارائه گردیده و نهایتاً در بخش ۵ نتیجه‌گیری می‌شود.

۲- پیشینه و انگیزه

تا به حال مطالعات زیادی برای کاهش مصرف انرژی در NoCها انجام شده است [۷]. سد توان یکی از روش‌هایی است که به طور گسترده در این زمینه مورد استفاده قرار گرفته که در آن منابع بیکار خاموش می‌شوند. مشکل این روش مواجه‌شدن با مسیریاب‌های خاموش در مسیر بسته و تأخیر ناشی از بیدارشدن آنهاست. این امر میزان توان ایستای مصرفی مسیریاب‌ها را کاهش داده و روی کارایی NoC تأثیر منفی دارد. مرجع [۸] برای از بین بردن تأخیر راه‌اندازی مسیریاب‌های خاموش، از ظرفیت خالی واسط شبکه استفاده می‌کند تا در طول زمان بیدارشدن مسیریاب‌ها بسته‌ها جابه‌جا و منتقل شوند. البته این روش، سربار مساحت زیادی را تحمیل می‌کند؛ زیرا نیاز به تعداد زیادی مدار انتخابگر و حافظه میانگیر دارد. در [۹] از نوع خاصی از NoCهای موازی که معماری NoC چندتایی نامیده می‌شود، استفاده گردیده است. در این معماری از روش سد توان استفاده شده که برای کاهش تأخیر، از تغییر زیرشبکه برای عدم روبه‌روشدن با مسیریاب‌های خاموش استفاده شده است.

مقیاس‌بندی ولتاژ، روش دیگری است که در بسیاری از معماری‌های شبکه بر تراشه به منابع شبکه اعمال می‌شود [۱۰]. در این روش، ولتاژ تغذیه بر اساس معیارهای مختلف تغییر داده می‌شود. این روش نیز تأخیر شبکه را افزایش می‌دهد و اصولاً کنترلی بر ازدحام شبکه ندارد؛ بنابراین توجه به میزان تأخیر شبکه، چالش مهمی برای مواردی است که از این روش استفاده می‌کنند. در [۱۱] نوسان ولتاژ چندسطحی ارائه شده تا مصرف انرژی را با هزینه از دست دادن کیفیت خروجی کاهش دهد. در این روش، ابتدا داده‌هایی که تحمل‌پذیری خطا دارند، توسط یک رابط مبتنی بر نرم‌افزار مشخص شده و بر اساس آن، روش تغییر ولتاژ منابع به ازای معیارهای مختلف اعمال می‌شود.

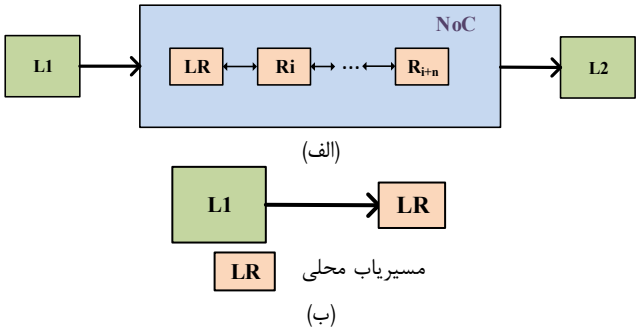
در [۱۲] و [۱۳] روش مقیاس‌بندی ولتاژ بر اساس میزان سازگاری در حافظه نهان برای کاهش مصرف انرژی اعمال شده است؛ اما این امر کارایی و قابلیت اطمینان سیستم را کاهش می‌دهد. در [۷]، منابع NoC برای تغییر ولتاژ منبع تغذیه مسیریاب‌ها و پیوندهای پرمصرف مجدداً پیکربندی شده‌اند؛ اما تغییر سطح ولتاژ، ازدحام را در شبکه افزایش می‌دهد و تأخیر به شبکه تحمیل می‌کند. برای جبران تأخیر تحمیل‌شده روی شبکه، میزان جابه‌جایی داده‌ها در طول شبکه باید کاهش یابد. این مسئله را می‌توان با رویکردهای مختلف انجام داد. نگاشت مناسب وظایف روی هسته‌ها، رویکردی است که نیاز به جابه‌جایی اطلاعات بین آنها را به حداقل می‌رساند [۵]. روش‌های مدیریت داده در حافظه، یکی دیگر از این رویکردهاست [۱۴].

معمولاً در MPSoC، به دلیل محدود بودن سطح تراشه، حافظه نهان سطح۲ به صورت توزیع‌شده می‌باشد و در نتیجه بخشی قابل توجهی از ترافیک شبکه ناشی از فقدان داده در این حافظه نهان توزیع‌شده است. برخی از پژوهشگران سعی کرده‌اند این ترافیک اضافی را با مدیریت حافظه نهان کاهش دهند. تکرار داده‌ها [۱۵]، مهاجرت داده‌ها [۱۶] تا [۱۸]، کنترل جریان داده [۱۹] و ذخیره موقت داده‌ها در بخش بیکار (خالی) منابع [۲۰]، نمونه‌هایی از روش‌های مدیریت داده‌ها هستند. وقتی شبکه در شرایط ازدحام نباشد، کانال‌های مجازی در حافظه میانگیرهای ورودی مسیریاب‌ها معمولاً خالی هستند. بر اساس نتایج حاصل از محک‌های مربوط به برنامه‌های کاربردی در سامانه‌های چندهسته‌ای، به طور متوسط کمتر از ۵ درصد از ظرفیت حافظه میانگیر کانال‌های مجازی استفاده می‌شود [۱۳]. هنگامی که فقدان در حافظه نهان در L۱ اتفاق می‌افتد، برای آوردن بلوک جدید باید یک بلوک از آن سطح حافظه نهان اخراج شود. این بلوک از طریق کانال مجازی مسیریاب محلی و ارسال روی شبکه به سطح بالاتر می‌رود. برای کاهش ترافیک ناشی از این انتقال در [۲۰]، از ظرفیت خالی میانگیرهای کانال‌های مجازی مسیریاب محلی برای ذخیره موقت و ارسال با تأخیر این بلوک‌های اخراج‌شده استفاده می‌شود و در صورت نیاز مجدد به این بلوک‌ها، آنها در مسیریاب محلی هستند و نیاز به انتقال مجدد از طریق شبکه نیست. از آنجا که نمی‌توان این بلوک‌ها را به صورت نامحدود نگهداری نمود، مهلت نگهداری این بلوک‌ها بر اساس دو سازوکار تحریک زمان ثابت (یعنی در یک زمان از پیش تعیین‌شده) و تحریک بر مبنای درخواست (یعنی به محض درخواست گره دیگر در شبکه) تعیین می‌شود.

در این مقاله، روش مقیاس‌بندی ولتاژ را به صورت تطبیقی و با چند سطح مختلف اعمال می‌کنیم. از آنجا که برای عملکرد بهتر مقیاس‌بندی ولتاژ باید ازدحام در شبکه کمتر باشد، ما نیز از ارسال بلوک‌های کثیف جایگزین‌شده روی شبکه پیشگیری نموده و بلوک خارج‌شده از حافظه نهان L۱ در میانگیر کانال‌های مجازی مسیریاب محلی را نگه می‌داریم. اما برای این منظور سازوکار جدیدی برای تعیین زمان این نگهداری موقت پیشنهاد نموده‌ایم که در آن، زمان نگهداری را بر اساس میزان فضای خالی میانگیرها در سه سطح مشخص می‌کنیم و آن را تحریک بر مبنای فضا نامیده‌ایم. بدیهی است که فضای خالی بیشتر در میانگیرها امکان نگهداری طولانی‌تر بلوک‌های اخراج‌شده را می‌دهد. به این ترتیب با کاهش میزان جابه‌جایی داده‌ها در شبکه می‌توان از روش مقیاس‌بندی ولتاژ بدون افت قابل توجه در کارایی شبکه استفاده نمود. شایان ذکر است که مقیاس‌بندی ولتاژ را نیز در سه سطح مختلف بر اساس فضای خالی میانگیر تنظیم نموده‌ایم. مثلاً هنگامی که حافظه میانگیر نزدیک به پر است، باید مقدار ولتاژ منبع تغذیه در مقدار بیشینه باشد. بدین ترتیب می‌توان به بهترین حالت ممکن در مصرف انرژی در شبکه دست یافت.

۳- مقیاس‌بندی ولتاژ تطبیقی پیشنهادی

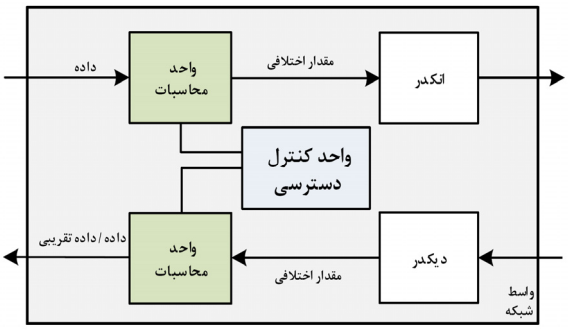
روش مقیاس‌بندی ولتاژ برای کاهش مصرف انرژی در NoCها به طور گسترده مورد استفاده قرار گرفته است. از آنجا که مقیاس‌بندی ولتاژ باعث افزایش نرخ خطا می‌شود، باعث ارسال مجدد بسته‌ها، ایجاد ازدحام و تحمیل هزینه تأخیر در شبکه نیز می‌شود. اعمال روش‌های مؤثر مدیریت ترافیک و کنترل ازدحام در شبکه باعث می‌شود تا بتوان از ظرفیت‌های مقیاس‌بندی ولتاژ برای کاهش به صورت بهتری بهره گرفت. در این مقاله،



شکل ۱: (الف) مسیر بسته خارج‌شده از حافظه نهان در NoC متداول و (ب) مسیر بسته خارج‌شده از حافظه نهان در NoC پیشنهادی.

روش مقیاس‌بندی ولتاژ را به صورت تطبیقی و در سه سطح ولتاژ مختلف با توجه به میزان ازدحام حافظه میانگیرهای ورودی مسیریاب شبکه‌های بر تراشه پیشنهاد داده‌ایم. برای این منظور، ابتدا میزان جابه‌جایی داده‌ها در شبکه را به وسیله روش ذخیره و فراخوانی پاسخ از حافظه میانگیر محلی مسیریاب کاهش می‌دهیم. در این روش داده خارج‌شده از حافظه نهان به جای عبور از مسیریاب‌های شبکه برای رسیدن به حافظه سطح ۲، در فضای خالی میانگیرهای مسیریاب محلی ذخیره می‌شود که زمان دسترسی مجدد به داده را کاهش داده و از ایجاد ترافیک غیرضروری در شبکه جلوگیری می‌کند. پس از اینکه ترافیک در سطح شبکه کاهش یافت، به منظور کاهش مصرف انرژی در شبکه از روش تغییر سطح ولتاژ منابع استفاده می‌شود. سپس ولتاژ منابع شبکه بین سه سطح مختلف تغییر می‌کند تا مصرف انرژی در شبکه تا جای ممکن کاهش یابد. ما از منابع بیکار شبکه به عنوان حافظه میانگیر نوشتن در سلسله‌مراتب حافظه استفاده می‌کنیم تا ازدحام را کاهش دهیم. Write Buufer یا حافظه میانگیر نوشتن، روشی متداول برای سیاست بازنویسی در سازگارنمودن سطوح حافظه نهان است [۲۱] که در آن بلوک‌های خارج‌شده در یک حافظه میانگیر نگهداری می‌شوند تا به تدریج به سطح بعدی حافظه ارسال شوند. نکته اصلی، سرعت متفاوت در دو طرف انتقال است که لایه فعلی را برای ادامه کارش آزاد می‌کند. گاهی اوقات قبل از خروج از حافظه میانگیر به بلوک‌های خارج‌شده مجدداً نیاز می‌شود. در این حالت بلوک خارج‌شده به لایه زیرین بازگردانده می‌شود؛ بنابراین میزان جابه‌جایی داده بین لایه‌های حافظه کاهش می‌یابد.

یک MPSoC به‌صورت کاشی‌هایی که توسط یک شبکه بر تراشه به هم متصل می‌شوند، سازماندهی شده است. هر کاشی از یک هسته پردازشی، یک حافظه نهان L۱، یک بانک حافظه نهان L۲ مشترک و یک واسط شبکه (NI) تشکیل شده است. هنگامی که فقدان داده در حافظه نهان L۱ رخ می‌دهد، بلوک درخواستی از حافظه نهان L۲ واکشی می‌شود. از آنجا که حافظه نهان L۲ بین گره‌ها مشترک است و در سراسر NoC توزیع می‌شود، بلوک درخواستی ممکن است از L۲ همان گره که دسترسی محلی نامیده می‌شود یا از حافظه نهان L۲ سایر گره‌ها که دسترسی غیرمحلی نامیده می‌شود، واکشی شود. این دسترسی غیرمحلی بسیار زمان‌بر است و میزان انرژی مصرفی زیادی را می‌طلبد. همچنین بلوک خارج‌شده باید به حافظه نهان L۲ مرتبط ارسال شود که ممکن است نیاز به دسترسی محلی یا راه دور شود. در صورت دسترسی از راه دور، بلوک خارج‌شده باید در سراسر شبکه جابه‌جا شود. حافظه میانگیر ورودی مسیریاب‌ها می‌تواند نقش حافظه میانگیر نوشتن را در این دو سطح حافظه ایفا کند. در واقع، برخی از این بلوک‌ها ممکن است به زودی دوباره درخواست شوند. از طرفی، بخشی از ظرفیت حافظه میانگیرهای



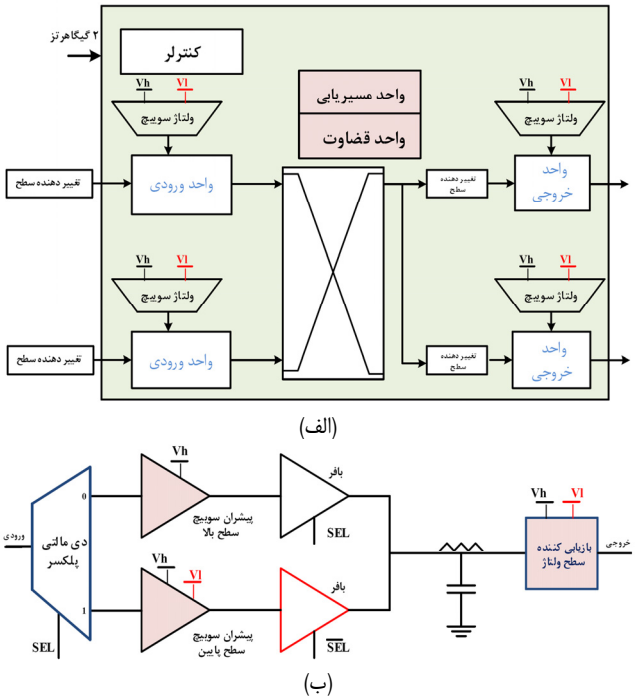
شکل ۲: معماری ذخیره و فراخوانی پاسخ محلی از شبکه.

NoC به‌جز در وضعیت ازدحام NoC معمولاً خالی است. ما از فضای خالی در کانال‌های مجازی مسیریاب‌های محلی برای ذخیره بلوک‌های حافظه نهان L۱ استفاده می‌کنیم. این روش ذخیره‌سازی از انتقال غیرضروری بسته‌ها در شبکه جلوگیری می‌کند. درخواست بعدی بلوک‌های خارج‌شده توسط مسیریاب محلی پاسخ داده می‌شود. رویکرد پیشنهادی ما، میزان جابه‌جایی داده‌ها در شبکه و جریمه عدم دسترسی به داده در حافظه نهان را کاهش می‌دهد که به معنای کاهش تأخیر شبکه است. از این رو امکان به‌کارگیری روش مقیاس‌بندی ولتاژ در پیوندها و مسیریاب‌های NoC بدون کاهش عملکرد در شبکه فراهم شده است. روند انجام رویکرد پیشنهادی در NoC در شکل ۱ نشان داده شده است. به این ترتیب جابه‌جایی داده‌ها را در شبکه بر اساس میزان محلیت داده‌های برنامه کاربردی کاهش می‌یابد و امکان کاهش مصرف انرژی با مقیاس‌بندی ولتاژ در شبکه فراهم می‌شود. برای استفاده حداکثری از ظرفیت ایجادشده، سطوح مختلف ولتاژ در منابع شبکه را با توجه به میزان اشغال حافظه میانگیرهای ورودی مسیریاب‌ها تنظیم می‌کنیم.

۳-۱ ذخیره و فراخوانی بلوک داده در/از مسیریاب محلی

بلوک‌های خارج‌شده از حافظه نهان باید مسیری را در شبکه بر تراشه طی کنند و در VCهای مسیریاب‌های مسیر ذخیره شوند تا به مقصد برسند. ما یک پرچم را به بلوک‌های خارج‌شده از حافظه نهان برای تشخیص این بلوک‌ها اختصاص می‌دهیم. همان طور که در شکل ۲ نشان داده شده است، هنگامی که بسته به کانال مجازی مسیریاب محلی می‌رسد، این پرچم توسط واحد دسترسی محلی (LAU) بررسی می‌شود. اگر مقدار این پرچم روی یک تنظیم شده باشد، تخصیص کانال مجازی و عملکرد سوئیچینگ برای بسته متوقف می‌شود. این بسته‌ها تا یک زمان آستانه تعیین‌شده بر اساس ازدحام شبکه در مسیریاب محلی باقی می‌مانند.

هنگامی که عدم دسترسی به داده حافظه نهان رخ می‌دهد، کنترل‌کننده حافظه نهان L۱ داده‌ها را از بانک حافظه نهان L۲ مربوطه درخواست می‌کند. بر اساس مکان حافظه نهان L۲ روی تراشه، کنترلر L۲ بلوک را به هسته درخواستی ارسال می‌کند. این روش برای ارسال داده‌ها با توجه به وضعیت ازدحام NoC یک مقدار زمانی را می‌طلبد. هنگامی که یک بسته در حافظه نهان یافت نشد، مسیریاب محلی بررسی می‌شود، LAU ابتدا حافظه میانگیرهای مسیریاب محلی را برای داده‌های درخواستی بررسی می‌کند. آدرس بلوک‌های درخواستی را با آدرس ذخیره‌شده در میانگیرهای اشغال‌شده مقایسه می‌کند. اگر مطابقت داشته باشد، LAU منبع و مقصد آن را با داده‌های درخواستی جایگزین می‌کند. وقتی هسته‌های دیگر به داده ذخیره شده در مسیریاب محلی هسته‌های دیگر نیاز دارد، یک بلوک کثیف حافظه نهان L۱ در مسیریاب محلی تا یک آستانه زمانی مشخص ذخیره می‌شود که این مقدار با توجه به زمان



شکل ۳: معماری منابع NoC با چند سطح ولتاژ (الف) مسیریاب با ولتاژ سه‌سطحی و (ب) ماژول پیوند با دو سطح ولتاژ.

ارجاع‌دهی مجدد بلوک‌های خارج‌شده تعیین می‌شود. یک شمارنده به هر کانال مجازی پورت ورودی اضافه شده است؛ وقتی شمارنده به مقدار آستانه رسید، بلوک ذخیره‌شده در کانال مجازی به واحدهای تخصیص سوئیچ و تخصیص کانال مجازی برای ارسال به مقصد فرستاده می‌شود. بنابراین زمانی که بلوکی به صورت محلی ذخیره شده است، درخواست هسته‌های دیگر به اندازه آستانه زمانی تعیین‌شده در بانک حافظه نهان L۲ به تأخیر می‌افتد.

۳-۲ کاهش نوسان ولتاژ در منابع NoC

از آنجا که میزان جابه‌جایی داده‌ها در شبکه با فراخوانی پاسخ از داده‌های ذخیره‌شده در مسیریاب محلی کاهش می‌یابد، می‌توان میزان ولتاژ پیوندها و مسیریاب‌ها را کاهش داد تا معیارهای توان در NoC را بهبود داد. به منظور به دست آوردن بهینه مصرف انرژی برای منابع شبکه، سه سطح ولتاژ با توجه به میزان فضای اشغال‌شده حافظه میانگیرهای ورودی مسیریاب به منابع شبکه اعمال می‌شود. معماری مسیریاب و پیوند پیشنهادی در شکل ۳ نشان داده شده است. از آنجا که کاهش سطح ولتاژ منابع سبب افزایش تأخیر در شبکه و ایجاد خطا در خروجی می‌شود، ابتدا بهتر است به وسیله روشی ترافیک شبکه کاهش یابد؛ سپس سطح ولتاژ برای بخش‌های خطاپذیر برنامه کاهش یابد. به دلیل کاهش ترافیک شبکه ناشی از ذخیره داده‌ها در مسیریاب محلی و ماهیت تحمل‌پذیری خطای برنامه، ولتاژ تغذیه منابع می‌تواند به دو مقدار پایین‌تر تغییر کند. به‌کارگیری توأمان این دو روش ارتباطی، مصرف انرژی را بدون هزینه عملکردی کاهش می‌دهد، زیرا ترافیک شبکه کاهش یافته و دیگر با اعمال روش مقیاس‌بندی ولتاژ، تأخیر در شبکه افزایش نمی‌یابد. شبه‌کد فراخوانی پاسخ محلی از بلوک ناموجود در حافظه نهان ذخیره‌شده در فضای خالی حافظه میانگیر در شکل ۴ آمده است.

ما از یک مسیریاب ولتاژ سه‌سطحی استفاده می‌کنیم که بر اساس استراتژی سوئیچینگ کرم‌چاله در این مقاله استفاده شده که دارای تأخیر کم است و همان طور که در شکل ۳- الف نشان داده شده است، ولتاژ

Algorithm 1: local-reply resource management pseudo code

Input: Status of virtual channels
Output: energy-saving data communication
Notations:
P_{int}: Packet entered through local input port
SA: Switch Allocation, VA: Virtual Channel Allocation
1: if P_{int} = = Evicted Dirty then
2: Disable SA & VA for P_{int}
3: elseif P_{int} = = Miss then
4: if P_{TB} [Addr] = P_{int} [Addr] then
5: send from P_{TB}
6: else Enable SA & VA for P_{int}
7: Deallocate VC
8: switch the voltage level Vh (1V) to Vl (0.8V)

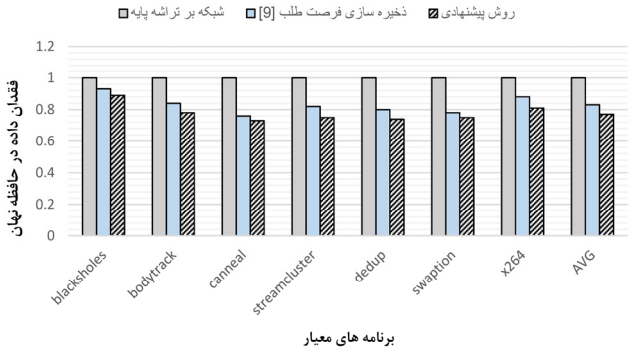
شکل ۴: شبه‌کد برای فراخوانی پاسخ محلی داده‌ها از مسیریاب.

بین مقادیر V_f، V_h و V_l تغییر می‌کند که در آن V_h حداکثر ولتاژی است که هیچ مقیاس ولتاژی روی گره‌ها اعمال نمی‌شود. کنترل‌کننده ولتاژ تعیین می‌کند از کدام سطح ولتاژ برای عبور از بسته استفاده می‌شود که آن هم با توجه به مقدار متوسط فضای آزاد حافظه میانگیرهای NoC تعیین می‌شود. ماژول پیوند در شکل ۳- ب نشان داده شده که در آن بر اساس مقدار متوسط فضای آزاد حافظه میانگیرهای NoC، انتخابگر (SEL) تنظیم شده است. اگر میانگین فضای آزاد حافظه میانگیر گره‌ها بیشتر از C۱٪ باشد، ولتاژ روی V_l و اگر بین C۱٪ و C۲٪ باشد، ولتاژ روی V_f تنظیم می‌شود. برای جلوگیری از ازدحام و تحمیل تأخیر به شبکه، زمانی که حافظه میانگیرها نزدیک به پر هستند، تکنیک مقیاس‌بندی ولتاژ متوقف می‌شود (ولتاژ روی V_h تنظیم می‌شود)؛ یعنی میانگین فضای آزاد حافظه میانگیر گره‌ها کمتر از C۲٪ باشد. مقدارهای این پارمترها بر اساس شبیه‌سازی انجام‌گردیده، تعیین شده است.

۳-۳ سازگاری در سطوح مختلف حافظه

برای حفظ سازگاری در حافظه، دو حالت را برای هر بلوک موجود در حافظه نهان سطح L۱ در نظر گرفتیم. حالت تمیز[1] تا زمانی است که فقط از آن بلوک خواندن انجام شده و حالت کثیف[2] که روی آن حداقل یک بار نوشتن انجام شده است. با اولین تغییر در بلوک‌های داخل حافظه نهان L۱ یک سیگنال نامعتبرسازی[3] به تمام حافظه‌های سطح ۱ در سایر گره‌ها ارسال می‌شود تا آن بلوک در آن سطح نامعتبر شوند؛ پس وقتی یک بلوک تغییریافته یا کثیف باشد، آن تنها نسخه موجود در آن سطح است. اگر یک بلوک کثیف موجود در گرهی توسط گره دیگری درخواست شود، بر اساس سازوکار آمده در بخش ۳-۱، ابتدا آن بلوک با یک تأخیر قابل تنظیم در سطح L۲ به‌روز شده و سپس به حافظه نهان L۱ گره درخواست‌کننده منتقل می‌شود و پس از این روند هر دو گره، نسخه‌های تمیز آن بلوک را دارند و سازگاری داده برقرار است. ذخیره موقت در میانگیرهای مسیریاب با سازوکار تحریک بر مبنای فضا که در این مقاله پیشنهاد شده است، برای بلوک‌های کثیف که در اثر جایگزین‌شدن باید به سطح L۲ انتقال یابند، انجام می‌شود و از آنجا که آنها تنها نسخه موجود در آن سطح هستند، در ذخیره موقت نیاز به انجام عمل خاصی برای حفظ سازگاری حافظه نیست. در زمان جایگزینی بلوک‌های تمیز حافظه نهان L۱ نیز اساساً نیازی به انتقال به سطح ۲ نیست و سازگاری برقرار است.

1. Clean Block
2. Dirty Block
3. Invalidation



شکل ۵: جریمه از دست دادن حافظه پنهان برای برنامه معیار PARSEC.

جدول ۱: پارامترهای شبیه‌سازی سیستم.

پارامتر	مقدار
تعداد هسته‌ها	۶۴ (x۸۶ ISA)
توپولوژی و اندازه شبکه	Mesh ۸×۸
اندازه حافظه نهان L۱	۳۲ KB ،۴-way
اندازه حافظه مشترک L۲	۱۶ MB ،۸-way
مقادیر V_l ،V_f و V_h	۰٫۸، ۰٫۹ و ۱ ولت
C۱ و C۲	۴۰٪ و ۷۰٪

جدول ۲: معیارهای ارزیابی کیفیت خروجی برنامه‌های معیار مختلف [۲۵].

نام	حوزه	معیار کیفیت	مجموعه پروفایل
Black-scholes	تحلیل مالی	درصد قیمت‌های با خطای بیشتر از ۱٪	۵۳۶۶۵ گزینه مالی با پارامترهای متعدد
Canneal	یادگیری ماشین	خطای نسبی مسیریابی نهایی	نت‌لیست ساختگی با ۴۰۰۰۰۰ نقطه
Ferret	جستجوی تصویر	درصد تصاویر جستجونشده در نتایج دقیق	۲۵۶ تصویر
Fluida-nimate	انیمیشن	درصد ذرات با خطای بیش از ۰٫۲٪ برای محل نهایی	۸۰۹۳۰۵ داده از ذرات نیوتنی
Stream-cluster	داده‌کاوی	درصد نقاطی که خوشه‌بندی دقیق نشده‌اند.	۸۱۹۲۱۶ نقاط بعدی
Vips	پردازش تصویر	خطای نسبی متوسط مقادیر پیکسل RGB خروجی	تصویر رنگی Vips ۱۶۶۲×۵۵۰۰

٤- ارزیابی و تجزیه و تحلیل نتایج

برای تجزیه و تحلیل روش ارتباطی پیشنهادی، تغییراتی روی ماژول Garnet [۲۲] در شبیه‌ساز GEM۵ [۲۳] انجام شده است. ما مجموعه محک PARSEC [۲۴] را به‌عنوان بار کاری برای شبیه‌سازی یک برنامه کاربردی چندرشته‌ای در NoC در نظر گرفته‌ایم. مشخصه‌های شبیه‌سازی در جدول ۱ ارائه شده است. ما مجموعه‌ای از برنامه‌های معیار ارتباط‌محور و حافظه‌محور را از PARSEC به کار می‌گیریم. همان طور که در جدول ۲ نشان داده شده است، Blackscholes ترافیک کمی دارد؛ اما Ferret ترافیک بالایی را به NoC تحمیل می‌کند. Canneal نیز به عنوان یک معیار حافظه فشرده، بیشترین درخواست حافظه را دارد. ما برای بررسی این دسته از ترافیک‌ها، در ارزیابی بیشتر به شبیه‌سازی آنها پرداخته‌ایم. روش پیشنهادی ما با NoC متداول و ذخیره‌سازی فرصت‌طلب که در [۲۰] ارائه شده، مقایسه گردیده است. ذخیره‌سازی فرصت‌طلب از فضای بیکار حافظه میانگیرهای مسیریاب NoC برای ذخیره بلوک‌های خارج‌شده استفاده می‌کند، اما نمی‌تواند به طور قابل توجهی مصرف انرژی را در NoC کاهش دهد؛ زیرا مقدار ولتاژ بر اساس میزان ازدحام حافظه میانگیرها و با توجه به بخش‌های تقریبی برنامه کنترل می‌شود. ما علاوه بر آنکه بلوک‌های حافظه نهان خارج‌شده را در VCهای مسیریاب ذخیره می‌کنیم، مقیاس ولتاژ به‌صورت چندسطحی را در منابع NoC برای بهبود مصرف انرژی اعمال می‌کنیم.

٤-۱ عدم دسترسی به داده در حافظه نهان

معیار دسترسی به داده در حافظه نهان، تعداد چرخه‌های مورد نیاز برای واکشی داده‌ها از حافظه است. این یک معیار مناسب برای نشان‌دادن اثربخشی سازوکار ذخیره داده‌ها در حافظه میانگیر مسیریاب است. جریمه فقدان داده در حافظه نهان L۱ در مقایسه با معماری‌های مختلف نام‌برده در شکل ۵ نشان داده شده است. روش پیشنهادی ما جریمه فقدان داده در حافظه نهان L۱ را به ترتیب در مقایسه با NoC پایه و ذخیره‌سازی فرصت‌طلب به ترتیب ۲۲ و ۶ درصد کاهش می‌دهد. روش ذخیره‌سازی فرصت‌طلب [۲۰]، داده‌ها را به مسیریاب محلی ارجاع می‌دهد تا تأخیر ارتباط را کاهش دهد؛ با این حال نمی‌تواند طول بسته ارسال‌شده در شبکه را کاهش دهد. در Canneal در مقایسه با سایر محک‌ها، جریمه فقدان داده در حافظه نهان کمتری را نشان می‌دهد؛ زیرا زمان ارجاع مجدد بزرگ‌تری به بلوک‌های حافظه نهان خارج‌شده L۱ را دارد.

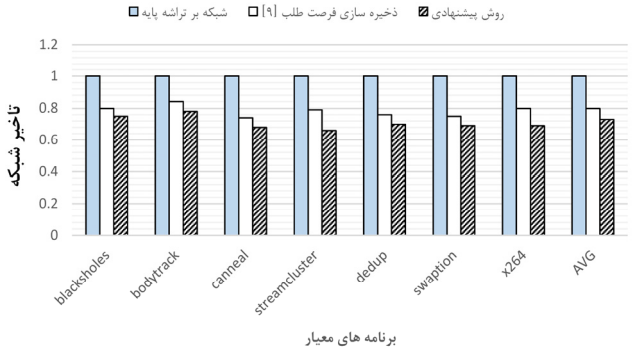
٤-۲ تأخیر شبکه

تأخیر شبکه، معیاری است که نشان می‌دهد چقدر از ازدحام شبکه کاهش یافته است. شکل ۶ مقدار نرمال‌شده تأخیر شبکه را نسبت به NoC پایه نشان می‌دهد. همان طور که نشان داده شده است، روش ما میانگین تأخیر شبکه را به ترتیب ۲۶ و ۹ درصد در مقایسه با حافظه پنهان NOC پایه و روش ذخیره‌سازی فرصت‌طلب کاهش می‌دهد. روش پیشنهادی ما به ترتیب ۲۵، ۲۲، ۳۲ و ۳۰ درصد کاهش تأخیر را در محک‌های blackscholes، bodytrack، canneal و x۲۶۴ در مقایسه با ذخیره‌سازی فرصت‌طلب به دست می‌آورد. شایان ذکر است که شبیه‌سازی‌ها در دو روش پیشنهادی و ذخیره‌سازی فرصت‌طلب به ازای شرایط برابر شامل اندازه و تعداد بسته یکسان صورت گرفته است.

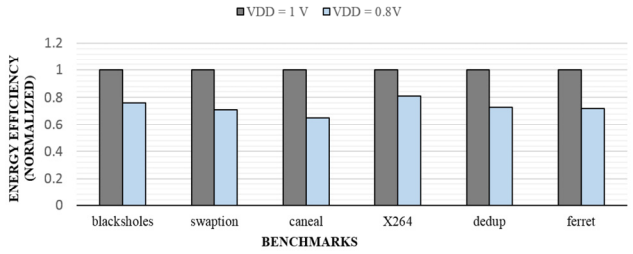
٤-۳ مصرف انرژی

هدف اصلی روش پیشنهادی، کاهش مصرف انرژی شبکه بر تراشه است. در این بخش، کارایی روش مقیاس‌بندی ولتاژ را از نظر مصرف انرژی NoC ارزیابی می‌کنیم. با توجه به نتیجه شبیه‌سازی نشان داده شده در شکل ۷ با کاهش ولتاژ منابع NoC از ۱ ولت به ۰٫۸ ولت، بازده انرژی ۲۵٪ بهبود می‌یابد. با این حال، زمانی که هیچ راه‌حلی برای کاهش ترافیک و زمان دسترسی به داده‌ها استفاده نمی‌شود، به میزان ۹٪ تأخیر به شبکه در مقایسه با NoC پایه تحمیل می‌شود.

برای نشان‌دادن کارایی و اثربخشی روش پیشنهادی، شبیه‌سازی‌ها برای مقیاس‌بندی ولتاژ دوسطحی و سه‌سطحی انجام شده است. مصرف



شکل ۶: تأخیر شبکه برای مجموعه معیار PARSEC.



شکل ۷: مقایسه مصرف انرژی در ولتاژهای مختلف تغذیه.

انرژی برای NoC پایه، روش ذخیره‌سازی فرصت‌طلب [۲۰] و روش پیشنهادی ما در دو حالت، یعنی مقیاس ولتاژ دوسطحی و سه‌سطحی در شکل ۸ آمده است. همان طور که مشاهده می‌شود، نسبت به NoC پایه، اعمال روش مقیاس‌بندی ولتاژ دوسطحی و سه‌سطحی مصرف انرژی را به طور میانگین به ترتیب ۲۷ و ۲۹ درصد بهبود می‌بخشد.

روش‌های دوسطحی و سه‌سطحی مقیاس‌بندی ولتاژ، مصرف انرژی را به ترتیب ۱۰ و ۱۲٫۵ درصد در مقایسه با روش ذخیره‌سازی فرصت‌طلب بهبود می‌بخشند. در مقایسه با NoC پایه، مصرف انرژی در روش منابع با ولتاژ سه‌سطحی برای برنامه‌های معیار blackscholes، bodytrack، canneal و x۲۶۴ به ترتیب ۲۹، ۲۴، ۳۴ و ۲۷ درصد کاهش می‌یابد.

٤-٤ کیفیت خروجی

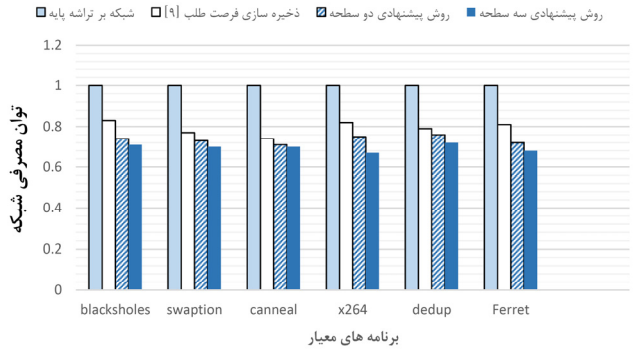
ما از معیارهای کیفیت مطابق جدول ۱ برای ارزیابی کیفیت خروجی استفاده می‌کنیم. شکل ۹ میزان کاهش کیفیت روش ارتباطی پیشنهادی را نشان می‌دهد. از آنجا که افت کیفیت ۷٪ به عنوان محدوده قابل قبول است [۲۵] و روش پیشنهادی، کیفیت خروجی را به طور متوسط ۵٫۷۵٪ کاهش می‌دهد، میزان افت کیفیت برای کاربران نهایی قابل قبول است.

٤-٥ سربار مساحت

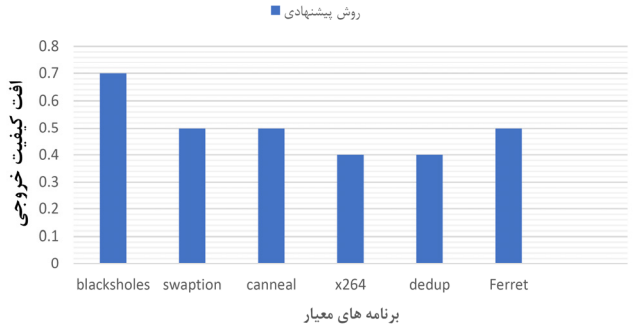
از Sinopsis Design Compiler برای تخمین مساحت و سربار معماری پیشنهادی در مدل فناوری ۴۵ نانومتری استفاده شده است. نتایج این امر نشان داد NI اصلاح‌شده، لینک و معماری مسیریاب در مقایسه با NoC اصلی مساحت فقط ۳٫۵٪ افزایش داشته است.

۵- نتیجه‌گیری

برنامه‌های کاربردی داده‌محور به دلیل ترافیک سنگینی که باید بین کاشی‌های MPSoCها منتقل شود، مصرف انرژی بالایی را به شبکه بر تراشه و سیستم تحمیل می‌کند. بخش قابل توجهی از این ترافیک از فقدان داده در حافظه نهان L۱ ناشی می‌شود که منجر به درخواست داده از حافظه نهان L۲ می‌شود که روی تراشه در کاشی‌های مختلف به اشتراک گذاشته می‌شود. بلوک‌های حافظه نهان L۱ خارج‌شده وارد



شکل ۸: مقایسه مصرف انرژی در مقیاس ولتاژ تطبیقی.



شکل ۹: کیفیت خروجی برای NoC پیشنهادی.

مسیریاب‌های محلی می‌شوند و از مسیریاب‌های زیادی در طول شبکه بر تراشه عبور می‌کنند تا در بانک حافظه نهان L۲ ذخیره شوند. در این مقاله از فضای خالی موجود در میانگیرهای مسیریاب‌های محلی با سازوکار جدیدی به نام تحریک بر مبنای فضا برای ذخیره بلوک‌های خارج‌شده از سطح ۱ حافظه نهان استفاده کردیم تا میزان جابه‌جایی داده‌ها و ازدحام در سطح شبکه کاهش یافته و به نوعی میزان فضای موجود در حافظه سطح ۲ افزایش یابد. پس از اینکه ترافیک در سطح شبکه کاهش یافت، به منظور کاهش مصرف انرژی در شبکه از روش تغییر سطح ولتاژ منابع استفاده می‌شود. بنابراین با طراحی مقیاس‌بندی ولتاژ تطبیقی با سه سطح ولتاژ مختلف که بر اساس میانگین ظرفیت فضاهای خالی در حافظه میانگیرهای مسیریاب محلی تنظیم می‌شود، مصرف انرژی را بهبود دادیم. شبیه‌سازی‌های متعدد نشان داد در مقایسه با روش مشابه ذخیره‌سازی فرصت‌طلب که روش مقیاس‌بندی ولتاژ منابع شبکه را در نظر نگرفته است، مصرف انرژی به میزان ۱۲٫۵٪ کاهش می‌یابد.

در ادامه این پژوهش می‌توان روی اعمال روش مقیاس‌بندی ولتاژ برای سایر روش‌های کاهش ترافیک روی تراشه، مانند جانشانی وظایف و/ یا روش‌های مدیریت داده مانند تکرار داده‌ها، انتقال داده‌ها و کنترل جریان داده متمرکز شد و میزان اثربخشی روش‌های کنترل ترافیک شبکه را روی عملکرد کلی شبکه و سیستم مطالعه نمود.

مراجع

[1] W. Amin, *et al.* "HyDra: hybrid task mapping application framework for NOC-based MPSoCs," *IEEE Access*, vol. 11, pp. 52309-52326, 2023.

[2] A. Kumar, N. Kumar, and B. Reddy, "An efficient real-time embedded application mapping for NoC based multiprocessor system on chip," *Wireless Personal Communications*, vol. 128, no. 4, pp. 2937-2952, 2023.

[3] S. P. Kaur, M. Ghose, A. Pathak, and R. Patole, "A survey on mapping and scheduling techniques for 3D network-on-chip," *J. of Systems Architecture*, vol. 147, Article ID: 103064, Feb. 2024.

[4] L. Mo, X. Li, A. Kritikakou, and X. Zhai, "Contention and reliability-aware energy efficiency task mapping on NoC-based

MPSoCs," *IEEE Trans. on Reliability*, vol. 74, no. 1, pp. 2010-2026, Mar. 2025.
[5] M. Momeni and H. S. Shahhoseini, "Energy efficient 3D network-on-chip based on approximate communication," *Computer Networks*, vol. 203, Article ID: 108652, 11 Feb. 2022.
[6] D. Deb and J. Jose, "ZPP: a dynamic technique to eliminate cache pollution in NoC based MPSoCs," *ACM Trans. on Embedded Computing Systems*, vol. 22, Article ID.: 118, 25 pp., 2023.
[7] M. Mineo, M. Palesi, G. Ascia, P. P. Pande, and V. Catania, "On-chip communication energy reduction through reliability aware adaptive voltage swing scaling," *IEEE Trans. Comput. Des. Integr. Circuits Syst.*, vol. 35, no. 11, pp. 1769-1782, Nov. 2016.
[8] Y. Ouyang, *et al.*, "DBU-PG: energy-efficient noc design using dual-buffering power gating," *the J. of Supercomputing*, vol. 80, pp. 13632-13656, 2024.
[9] M. Baharloo, R Aligholipour, M. Abdollahi, and A. Khonsari, "ChangeSUB: a power efficient multiple network-on-chip architecture," *Computer Electronic Engineering*, vol. 83, Article ID: 106578, May 2020.
[10] F. Yazdanpanah and R. A. Afsharmazayejani, "Systematic analysis of power saving techniques for wireless network-on-chip architectures," *J. of Systems Architecture*, vol. 126, Article ID: 102485, May 2022.
[11] G. Ascia, V. Catania, S. Monteleone, M. Palesi, D. Patti, J. Jose, and V. M. Salerno, "Exploiting data resilience in wireless network-on-chip architectures," *ACM J. Emerging Technology Computer System*, vol. 16, no. 2, Article ID:21, 27 pp., 2020.
[12] R. Hesse and N. E. Jerger, "Improving DVFS in NoCs with coherence prediction," in *Proc. of the 9th International Symposium on Networks-on-Chip*, Article ID:24, 8 pp., Vancouver, Canada, 28-30 Sept. 2015.
[13] T. Krishna, J. Postman, C. Edmonds, L. S. Peh, and P. Chiang, "SWIFT: a swing-reduced interconnect for a token-based network-on-chip in 90 nm CMOS," in *Proc. of IEEE Int. Conf. on Computer Design: VLSI in Computers and Processors*, pp. 439-446, Amsterdam, Netherlands, 3-6 Oct. 2010.
[14] R. Hesse, J. Nicholls, and N. E. Jerger, "Fine-grained bandwidth adaptivity in networks-on-chip using bidirectional channels," in *Proc. of IEEE/ACM 6th In. Symp. on Networks-on-Chip*, pp. 132-141, Lyngby, Denmark ,9-11 May 2012.
[15] N. Hardavellas, M. Ferdman, B. Falsafi, and A. Ailamaki, "Reactive NUCA: near-optimal block placement and replication in distributed caches," in *Proc. of the 36th Annual Int. Symp. on Computer Architecture*, pp. 184-195, Jun. 2009.
[16] G. Chen, F. Li, S. W. Son, and M. Kandemir, "Application mapping for chip multiprocessors," in *Proc. of the 45th Annual Design Automation Conf.*, pp. 620-625, Austin, TX, USA, 20-24 Jun. 2008.
[17] M. E. Wolf and M. S. Lam, "A data locality optimizing algorithm," *ACM SIGPLAN Notices*, vol. 26, no. 6, pp. 30-44, May 1991.
[18] U. Bondhugula, *et al.*, "Towards effective automatic parallelization for multicore systems," in *Proc. IEEE Int. Symp. on Parallel and Distributed Processing*, 5 pp., Miami, FL, USA, 14-18 Apr. 2008.
[19] F. Rad, M. Reshadi, and A. Khademzadeh, "Flow control and scheduling mechanism to improve network performance in wireless NoC," *IET Communications*, vol. 14, no. 14, pp. 2231-2239, Aug. 2020.
[20] A. Das, A. Kumar, J. Jose, and M. Palesi, "Opportunistic caching in NoC: exploring ways to reduce miss penalty," *IEEE Trans. on Computer*, vol. 70, no. 6, pp. 892-905, Jun. 2021.
[21] P. G. Massas and F. Pétrot, "Comparison of memory write policies for NoC based multicore cache coherent systems," in *Proc. of the Conf. on Design, Automation and Test in Europe*, pp. 997-1002, Mar. 2008.
[22] N. Agarwal, T. Krishna, L. S. Peh, and N. Jha, "GARNET: a detailed on-chip network model inside a full-system simulator," in *Proc. of the International Symposium on Performance Analysis of Systems and Software*, pp. 33-42, Munich, Germany, 10-14 Mar. 2009.
[23] N. Binkert, *et al.*, "The Gem5 simulator," *ACM SIGARCH Computer Architecture News*, vol. 39, no. 2, pp. 1-7, Aug. 2011.
[24] C. Bienia, S. Kumar, J. P. Singh, and K. Li, "The PARSEC benchmark suite: characterization and architectural implications," in *Proc. of the 17th. Parallel Architectures and Compilation Techniques*, pp. 72-81, Toronto, Canada, 25-29 Oct. 2008.
[25] J. San Miguel, M. Badr, and N. E. Jerger, "Load value approximation," in *Proc. of the 47th Annual IEEE/ACM Int. Symp. on Microarchitecture*, pp. 127-139, Cambridge, UK, 13-17 Dec. 2014.

سیده معصومه مومنی در سال ۱۳۹۰ مدرک کارشناسی مهندسی برق و در سال ۱۳۹۲ مدرک کارشناسی ارشد مهندسی برق را از دانشگاه گیلان دریافت نموده است. وی در حال حاضر دانشجوی دکتری دانشکده مهندسی برق دانشگاه علم و صنعت ایران است و زمینه‌های علمی مورد علاقه وی شبکه‌های روی تراشه و محاسبات تقریبی می‌باشد.

هادی‌شهریار شاه‌حسینی در سال ۱۳۶۷ مدرک کارشناسی مهندسی برق – الکترونیک خود را از دانشکده فنی دانشگاه تهران و در سال ۱۳۷۲ مدرک کارشناسی ارشد مهندسی برق – الکترونیک را از دانشگاه آزاد تهران و مدرک دکترای مهندسی برق در سال ۱۳۷۸ از دانشگاه علم و صنعت ایران دریافت نمود. وی از همان سال ۱۳۷۸ به عنوان عضو هیأت علمی در دانشکده مهندسی برق دانشگاه علم و صنعت ایران مشغول به فعالیت می‌باشد. زمینه‌های علمی مورد علاقه نام‌برده پردازش موازی و معماری کامپیوتر، مدیریت شبکه‌های کامپیوتری، یادگیری ماشین و الگوریتم‌های تکاملی و هوشمند می‌باشد.