



ارائه شده توسط:

سایت ترجمه فا

مرجع جدیدترین مقالات ترجمه شده

از نشریات معتبر

ElastiStore: معماری بافر الاستیک برای مسیریاب های شبکه بر روی تراشه

چکیده

طراحی معماری شبکه بر روی تراشه (NOC) مقیاس پذیر، خواستار پیاده سازی های جدیدی است که دستیابی به توان بالا و عمل با زمان تاخیر کم را بدون تجاوز از محدودیت های مساحت-انرژی سختگیرانه سیستم بر روی تراشه مدرن (SOC) ارائه می دهد. معماری بافر مسیریاب یکی از جنبه های طراحی مهم است که روی عملکرد و پیاده سازی ویژگی های شبکه گسترده ای تاثیر می گذارد. در این مقاله، ما معماری بافر الاستیک (EB) برای پشتیبانی از چندین کانال مجازی (VC) را گسترش می دهیم و ما ElastiStore، معماری بسیار سبک وزن بافر الاستیک جدید را بررسی می نماییم که الزامات بافر را بدون به خطر انداختن عملکرد به حداقل می رساند. ادغام طرح بافر الاستیک ارائه شده در مسیریاب NoC را برای طراحی معماری جدید مسیریاب - تک چرخه و دو مرحله پایپ لاین میسر می سازد - که ارائه دهنده عملکرد مشابه مسیریاب مبتنی بر پایه VC است که البته به طور قابل توجهی هزینه مساحت / توان پایین تری دارد.

TarjomeFa.Com

1. مقدمه

تکنولوژی شبکه بر روی تراشه در حال حاضر در اکثر SOC های بزرگ برای تسهیل یکپارچه سازی سیستم ها در سطح بررسی عملکرد IP-مونتاز - توسط کاهش ازدحام مسیریابی فیزیکی و بسته شدن ساده زمان اتخاذ می شود [1]. NOCS که همچنین عملکرد های موازی سازی ارتباطات را بهبود می بخشد، ارائه دهنده تضمین کیفیت خدمات (QoS)، و فعال کردن پارتیشن بندی سیستم قابل انعطاف است. اکثریت این ویژگی های NOC را می توان با استفاده از کانال های مجازی (VC های) برآورده نمود. کانال فیزیکی را می توان به شیوه زمان تسهیم شده توسط VC های مختلف استفاده نمود، به شرطی که هر VC صاحب یک فضای بافر جداگانه باشد [2]. معماری مبتنی بر VC- جداسازی ترافیک را اختصاص طبقات مختلف ترافیک به VC های مختلف میسر می سازد و آنها را ازدحام

مسیریابی فیزیکی بر روی تراشه را با معامله کردن عرض کانال فیزیکی و تعداد VC های پشتیبانی شده کاهش می دهند در نتیجه، یک طرح بندی انعطاف پذیر معماری SOC را ایجاد می نمایند [3].

NOC نیاز به مقیاس پذیر بودن، از نظر عملکرد شبکه و عملکرد، و همچنین انعطاف پذیر بودن از نظر پیاده سازی فیزیکی دارد. این نیاز به ما انگیزه متحد کردن معماری بر اساس VC را می دهد به نفع مقیاس پذیری NOC با بافر الاستیک است که پیاده سازی های فیزیکی و کاهش مساحت و توان را آسان می کند.

با توجه به عملیات الاستیک آن، که بر اساس دست دادن ساده آماده / معتبر است، بافر الاستیک یک شکل ابتدایی و ساده از بافر NOC است که می تواند به راحتی به شیوه ای پلاگین و بازی در ورودی و خروجی مسیریاب یکپارچه شود (و یا در داخل آنها) [4]، [5]، و همچنین در لینک های شبکه به عنوان یک تکرارکننده بافر شده عمل کند. بافرینگ الاستیک فقط یک شکل از دست دادن در هر یک از کانال های شبکه را فرض می کند که نمی تواند بین جریان های مختلف تمایز قائل شود که در نتیجه سریال فعالیت آن ساخته می شود. این ویژگی از ترک کردن بسته ها و جداسازی جریان ترافیک جلوگیری می کند، در حالی که پیشگیری از بن بست را پیچیده می کند. با توجه به این محدودیت، پشتیبانی مستقیم برای VC های رها شده است و با شبکه های فیزیکی متعدد، و یا اجرای ترکیبی پیچیده و معماری بافر غیر مقیاس پذیر [6] EB / VC، [7]، [8] جایگزینی می شود که ویژگی پایه EB ها را به عنوان عناصر دوقت حذف می کند که می تواند به صورت یکپارچه در هر نقطه از NOC قرار داده شود.

در این مقاله، ما عملیات و اجرای بافر الاستیک را برای حمایت از VC های متعدد تعمیم می دهیم. معماری ارائه شده، که ما ElastiStore می نامیم، تعداد بافرها در هر کانال (فلیپ فلاپ یا لچ را با توجه به پیاده سازی) نزدیک به حداقل مطلق یک حافظه بافر در VC، بدون به خطر انداختن عملکرد به حداقل می رساند. هر دو تک چرخه و دو مرحله پایپ لاین - - که عملکرد مشابه به عنوان مسیریاب مبتنی بر پایه VC را البته در یک مساحت به طور قابل توجهی پایین تر از مقیاس پذیری طرح ارائه شده توسط ادغام ElastiStore در مسیریاب NOC ارائه می دهند منجر به معماریهای جدید می شوند.

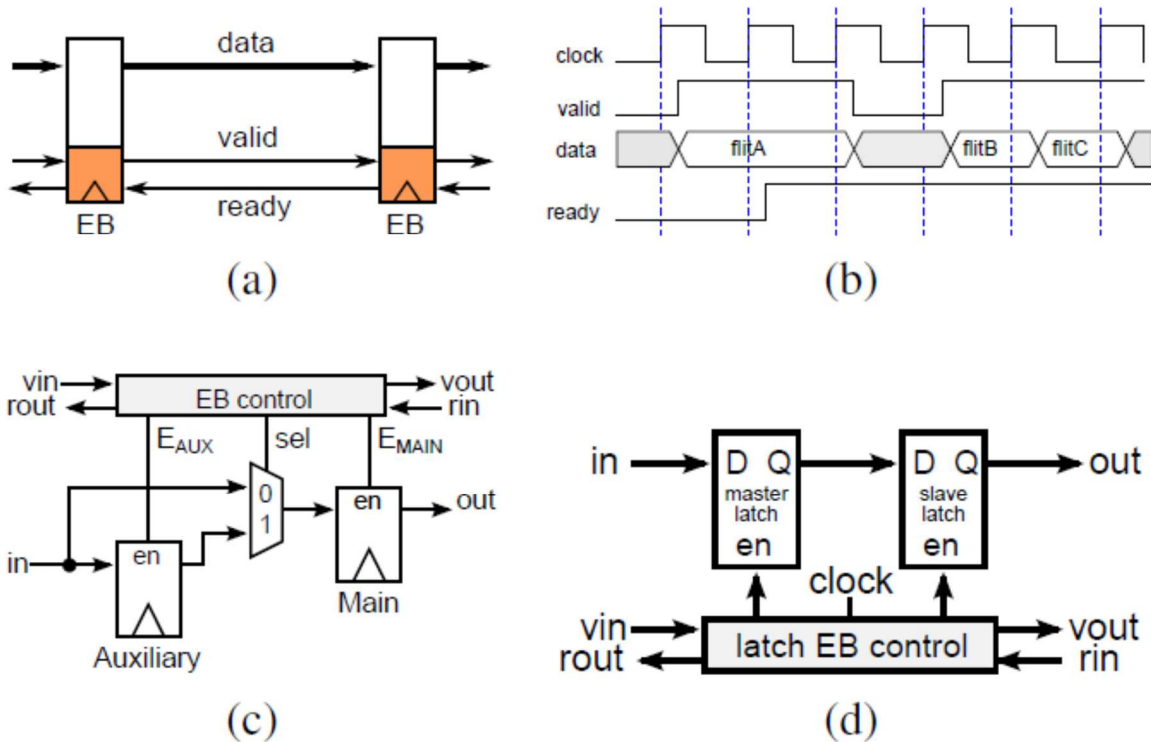
طرح پیشنهادی ElstiStore به صورت ابتدایی برای آینده، پیاده سازی مسیریاب NoC با هزینه بسیار کم، پیش بینی می شود که در آن عملکرد و قابلیت پیشرفت ارائه شده توسط VCها نمی تواند قربانی پیش بینی نمی شود. در واقع، به دلیل محدودیت های پروتکل پشتیبانی، استفاده از VCها در آینده تراشه چند پردازنده (CMP) با استفاده از دایرکتوری های مبتنی بر ذخیره سازی پروتکل های انسجام اجباری خواهد بود. این پروتکل انسجام نیاز به جداسازی بین طبقات پیام های مختلف، به منظور اجتناب از بن بست در سطح پروتکل دارد. به عنوان مثال، MOESI بر اساس دایرکتوری پروتکل وابستگی کش نیاز به حداقل سه شبکه مجازی برای جلوگیری از بن بست در سطح پروتکل دارد. شبکه مجازی شامل یک VC (و یا یک گروه از VCهای) با وظیفه دست دادن به یک کلاس پیام خاص از پروتکل انسجام حافظه نهان [9] دارد.

با وجود افزایش مطالبات عملکرد، بودجه مساحت / توان مسیریاب NoC فردی با حفظ تعداد عناصر پردازش (و، از این رو، به اندازه NoC) ادامه خواهد یافت. هدف از راه حل ElastiStore آشتی دادن دقیق خواسته های متضاد و واگرای عملکرد کم هزینه و با بالا است.

بقیه مقاله به شرح زیر است: بخش II به طور خلاصه شرح کنترل جریان الاستیک است، در حالی که بخش III معماری ElastiStore را معرفی می کند. بخش IV و V توصیف اجرای ElastiStore و ادغام آن در مسیریاب NoC است. نتایج آزمایش ها در بخش VII ارائه شده است، و نتیجه گیری در بخش VIII ترسیم شده است.

II. کانال پایه الاستیک و بافرها

کانال الاستیک پایه - به موازات سیم داده ها - دو سیم کنترل اضافی (معتبر و آماده) را حمل می کند که برای پیاده سازی پروتکل الاستیک مورد نیاز هستند، همانطور که در شکل 1 (الف) نشان داده شده است. EBها، پیاده سازی پروتکل الاستیک را با جایگزین کردن هر گونه ارتباط داده ساده با یک کانال الاستیک انجام می دهند.



شکل 1. پایه های پروتکل بافرینگ الاستیک

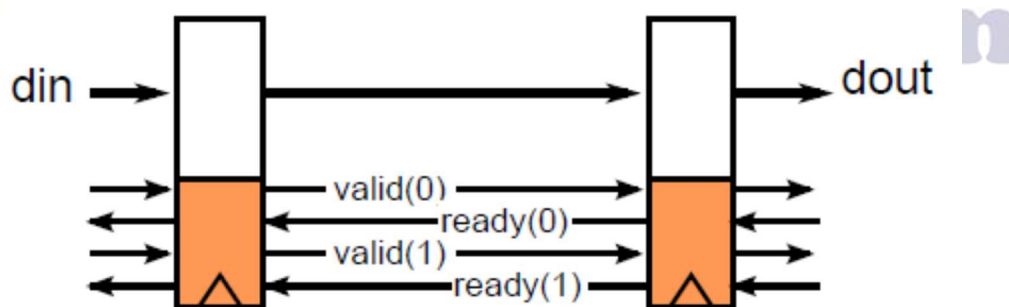
هنگامی که یک EB بتواند یک ورودی را بپذیرد، سیگنال آماده بالادست آن را ادعا می کند. هنگامی که دارای خروجی در دسترس است، سیگنال معتبر پایین دست را ادعا می کند. هنگامی که دو EB مجاور ببینند که سیگنال های معتبر و آماده هر دو درست هستند، آنها به طور مستقل می دانند که انتقال رخ داده است، بدون مذاکره و یا تصدیق. یک نمونه از این مورد در شکل 1 (ب) نشان داده شده است.

هنگامی که خروجی زنجیره ای EBها ممانعت شوند، این ممنعت فقط می تواند در یک مرحله در هر چرخه انتشار یابد. برای رسیدگی به این مورد، همه EBها می توانند دو واژه را نگهدارند، یکی برای خروجی متوقف شده و واژه گرفتار شده در صورت لزوم از مرحله قبل. این پیاده سازی در شکل 1 (ج) نشان داده شده است. EB 2 حافظه ای می تواند در سه حالت امکان پذیر باشد: EMPTY، HALF، و FULL، بسته به تعداد نقل مکان هایی که ذخیره شده است. با کنترل فاز ساعت بر این اساس، همانطور که در [10] نشان داده شده، EB 2 حافظه ای را می توان با

استفاده از 2 لیچ به صورت سری، به جای دو PAL طراحی نمود، که شبیه به شکل 1 (د) است. پس از روش مشابه، هر معماری EB استنتاج شده برای فلیپ فلاپ تحریک شده با لبه می تواند با لیچ اجرا شود.

III. VC های الاستیک

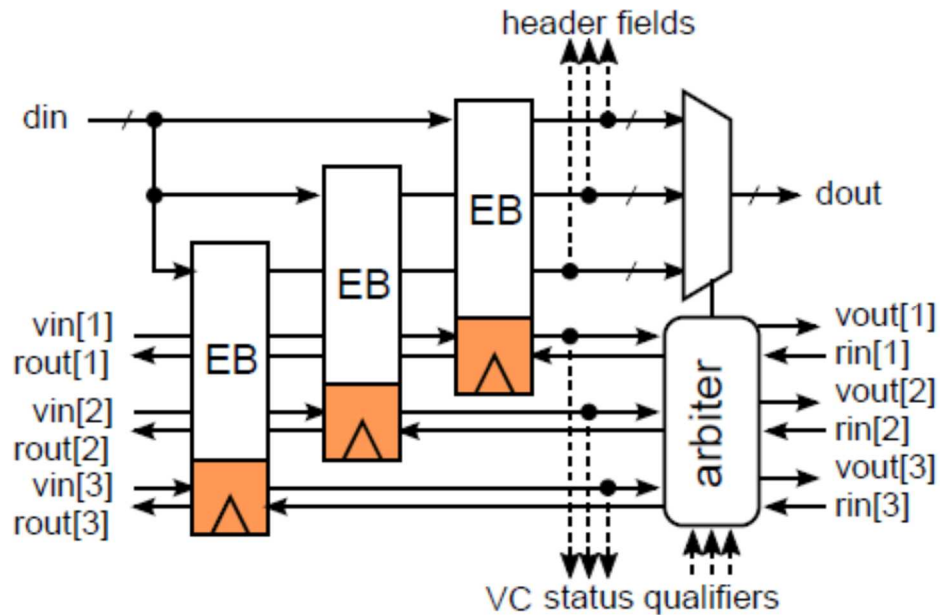
کانال الاستیکی که از VCها پشتیبانی می کند متشکل از مجموعه ای از سیم های داده است که انتقال یک نقل مکان را در هر سیکل کلاک انجام می دهد و همانند بسیاری از زوج های کابل کنترل معتبر (من) / آماده (من) با تعداد VCها. شکل 2 نمونه ای از یک کانال الاستیک VC-2 را نشان می دهد. در سطح پروتکل این روش شبیه به موضوعات مختلف [11 OCP-IP] است. از آنجا که VCهای متعدد می توانند در فرستنده فعال باشند، داوری برای انتخاب VC به کار گرفته می شود که از کانال استفاده می کند. در نتیجه، تنها سیگنال معتبر (i) در هر چرخه تأکید می شود. در همان زمان، گیرنده آماده پذیرش نقل مکان هایی است که به طور بالقوه می تواند به هر VC تعلق داشته باشد. بنابراین، هیچ محدودیتی در مورد تعداد آماده (د) سیگنال برای ادعا در هر چرخه وجود ندارد. داور در فرستنده باید تنها VC را می دهد که در گیرنده آماده است. بنابراین، درخواست های VCهای فعال برای اولین بار توسط سیگنال های آماده وارده واجد شرایط می شوند.



شکل 2. نمونه ای از کانال الاستیک VC-2

پایه ElastiStore اولیه را می توان با تکرار یک EB در VC، از جمله یک داور و یک مولتی پلکسر ساخت و پس از آن اتصالات در شکل 3 برای مورد VC 3 نشان داده شده است. داور انتخاب می کند که کدام VC خروجی را با چک کردن تزریق نماید اگر دارای داده های معتبر باشد و در صورتی که VC مربوطه آماده باشد. همچنین، مسابقات

مقدماتی اختیاری VC (1 بیت در VC) می تواند درخواست یک VC فعال را فعال یا غیر فعال نماید. این مسابقات انتخابی زمانی نیاز می شوند یک ElastiStore در ورودی مسیریاب NoC یکپارچه شود.



شکل 3. معماری ElastiStore اصلی

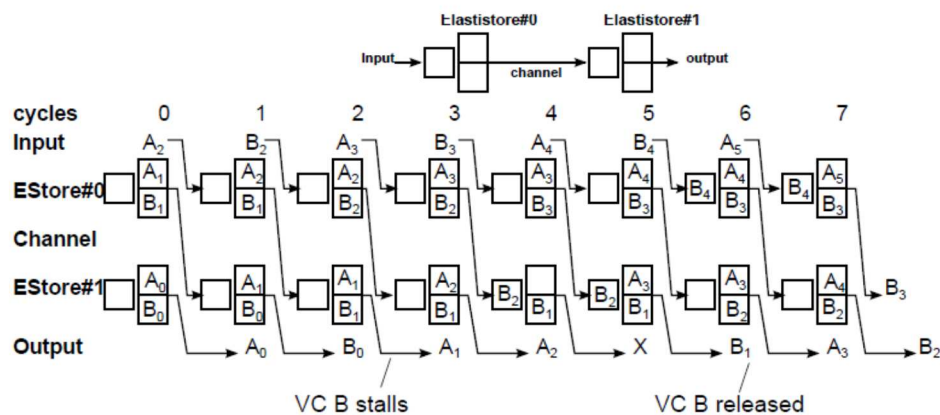
یک EB دو حافظه ای در هر VC اجازه می دهد تا هر یک از VC میزبان 2 نقل مکان باشد. این راه حل گران است، چرا که منابع در دسترس (در این مورد EBS) در هر VC تکرار می شوند. هنگامی که یک VC تنها بین دو پایه ElastiStores فعال است، می تواند به 100٪ توان دست یابد. در این حالت استفاده یکنواخت، هر VC فقط از یکی از دو بافر موجود در VC استفاده می کند، زیرا هر M چرخه یک بار در دسترس است. بافر دوم تنها زمانی که VC تاخیر دارد استفاده می شود.

دو بافر در هر VC به منظور میسر نمودن استفاده VC فعال از توان کامل نیاز است، حتی اگر بقیه VC $V-1$ های مسدود می شوند، که $2(V-1)$ بافر آن را اشغال می کند، و با فرض اینکه که هر یک از ذخایر VC ها مسدود شده درست برای راه اندازی مجدد در نرخ اوج باشند. در این مقاله، ما این محدودیت سخت را سبک می کنیم و ElastiStore را تنها با استفاده از $V-1$ بافر می سازیم. هر VC صاحب یک بافر تک (V در مجموع)، به اندازه کافی

در مورد استفاده یکنواخت است که در آن هر VC توان $M1 /$ را با $2 \leq M \leq V$ دریافت می کند. علاوه بر این ، هنگامی که یک VC تنها از کانال بدون هیچ گونه VC فعال دیگر و یا مسدود شده استفاده می کند، به عنوان مثال، $M = 1$ ، خروجی کامل را دریافت می کند و ، در مورد تاخیر، می تواند از بافر اضافی موجود در **ElastiStore** استفاده کند. این بافر اضافی به صورت پویا توسط تمام VC های به اشتراک گذاشته می شود، اگر چه فقط می توان آن را در هر چرخه ساعت داشت.

شکل 4 یک نمونه از جریان نقل مکان در یک کانال الاستیک که 2 VC را پشتیبانی می کند به تصویر می کشد . در ابتدا، همه VC ها در هر مرحله دارای یک نقل مکان در دسترس هستند. در اولین دوره ، هر VC $2/1$ توان را در هر کانال ($M = 2$) دریافت می کند و در هر مرحله ، آنها تنها از یک حافظه بافر استفاده می کنند. در آن دوره ، ثبات های کمی به اشتراک گذاشته شده ، استفاده نمی شوند. بافر مشترک بین 4 و 7 چرخه برای جای دادن واژه متوقف شده از VC B استفاده می شوند. در آن چرخه ، - VC A که مسدود شده است - همچنان به ارائه کلمات آن به خروجی کانال ادامه می دهد.

با این حال، زمانی که همه VC ها، به جز یکی ، مسدود می شوند، و بافر به اشتراک گذاشته شده توسط VC مسدود شده مورد استفاده قرار می گیرد، فقط فعال VC ، 50٪ از توان را دریافت می کند، زیرا به طور موثر تنها یک بافر موجود در هر کانال را می بیند. **ElastiStore** پایه و "گران" که 2 بافر را به هر VC اختصاص اجازه می دهد، این VC فعال را برای استفاده از کانال کامل مجاز می سازد.



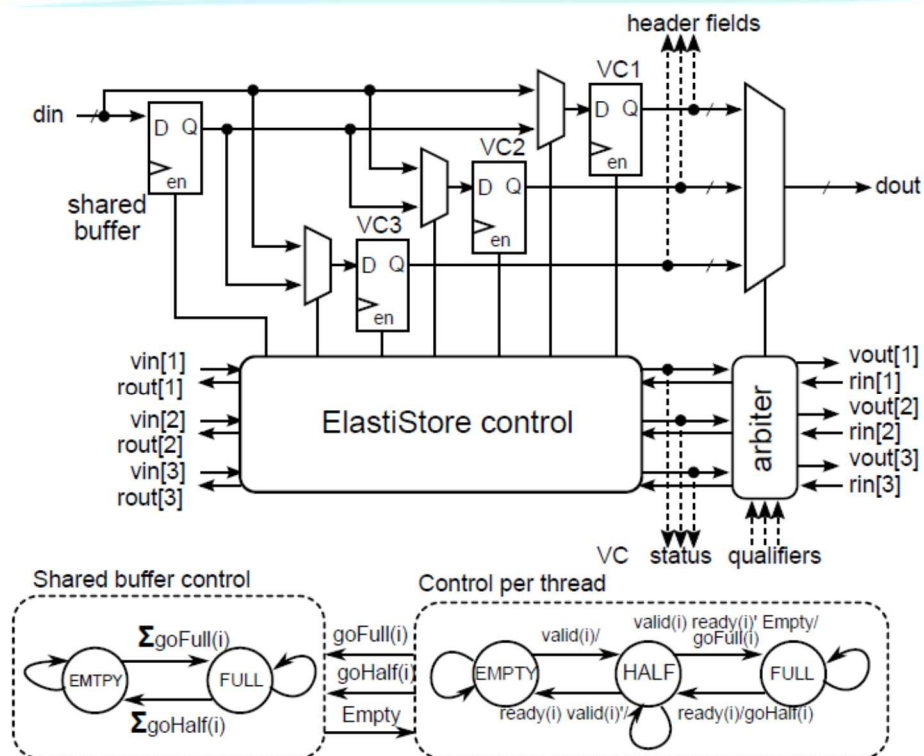
شکل 4. نمونه ای جریان نقل مکان روی کانال الاستیک که از 2 VC حمایت می کند.

ElastiStore ارائه دهنده یک موازنه منطقی است، زیرا موجب صرفه جویی در 1- V حافظه بافر در بافر VC الاستیک می شود و توان عملیاتی را فقط تحت ازدحام محدود می کند که تمام VC ها را به جز یکی مسدود می کند.

در مورد ترافیک سبک، فعال VC واحد عملیاتی کامل بدون هیچ محدودیت دریافت می کند. همچنین، تخصیص شخص از یک بافر تک به هر VC را تضمین پیشرفت رو به جلو برای همه VC های و جلوگیری ممکن است بن بست در سطح پروتکل.

IV. اجرای ELASTISTORE

ElastiStore را می توان با استفاده از مسیر داده نشان داده شده در شکل 5، که متشکل از ثبات تنها در VC همراه با ثبات به اشتراک گذاشته شده است طراحی نمود که به صورت پویا توسط تمام VC ها به اشتراک گذاشته می شود. سیگنال های انتخاب مالتی پلکسرهایی بای پس، بار سیگنال ثبات ها را قادر می سازد و همچنین به عنوان سیگنال های آماده / معتبر داخلی که به داور و رابط ورودی متصل هستند از طریق کنترل ElastiStore تولید می شوند.



شکل 5. مسیر داده و منطق کنترل معماری ارائه شده FSMS . ElastiStore منطق کنترل در پایین شکل نشان داده شده است.

کنترل ElastiStore ، V بار منطق کنترل EB تنها را کپی می کند که FSM 3 حالت نشان داده شده در شکل 5، را پیاده سازی می نماید که هر VC را برای بودن در حالت های EMPTY ، HALF ، و یا FULL میسر می سازد. ElastiStore کنترل حالت هر EB را با بررسی سیگنال های اضافی goFull و goHalf ردیابی می کند و از طریق سیگنال خروجی Empty تضمین می کند که تنها یکی از آنها به حالت FULL حرکت می کند. این مورد نیاز است ، چون فقط یک VC مجاز به ذخیره سازی دو نقل مکان است - در مورد توقف پایین دست - با استفاده از بافرهای به اشتراک گذاشته. یک FSM مرتبط با بافر به اشتراک گذاشته، این وضعیت را دنبال می کند و سیگنال های Empty را تولید می کند که تنها انتقال یک کنترل EB از حالت HALF به FULL را میسر می سازد.

هنگامی که یک کلمه جدید به ورودی می رسد و متعلق به VC i که در حالت Empty است ، در ثبات اصلی از VC i ذخیره می شود و به حالت HALF حرکت می نماید. در مقابل، VC های حالت HALF آماده برای پذیرش اطلاعات جدید می باشد ، تا زمانی که هیچ موضوعی در حالت FULL نباشد. اگر این مورد برقرار باشد، و داده های جدید برسند، سه عملیات در همان چرخه رخ می دهد : (الف) کلمه داده جدید در بافر مشترک ذخیره می شود، (ب) حالت آن به FULL حرکت می کند و (ج) تمام موضوعاتی که در حالت HALF متوقف هستند، آماده پذیرش داده های جدید هستند.

وقتی داور یک VC را انتخاب می کند که در حالت HALF است، داده های آن از ثبات های اصلی VC در صف بندی می شود و به حالت Empty باز می گردد. در مقابل، اگر VC انتخاب شده تنها در حالت FULL باشد - با دو واژه در ElastiStore ذخیره شود (در ثبات اصلی و بافر مشترک) - باید به حالت HALF حرکت کند ، بعد از اینکه خواندن داده ها از ثبات اصلی صورت می گیرد. در طی این انتقال حالت ، ثبات اصلی VC باید توسط داده های ذخیره شده در بافر مشترک پر شود . بافر مشترک نمی تواند یک کلمه جدید را در همان چرخه دریافت کند، زیرا در

دسترس بودن آن - که باعث آزاد شدن همه VC ها در حالت HALF می شود- در کانال بالادست در چرخه ساعت ظاهر خواهد شد.

حتی اگر مسیر داده از ElastiStore فقط به $V + 1$ ثبات کاهش یابد، کنترل آن تقریباً همان استفاده از EB 2 حافظه در VC است، همانطور که در شکل 3 نشان داده شده است. با استفاده از این خاصیت ما می توانیم طراحی ElastiStore مبتنی بر لچ را تنها با تغییر FSM در هر EB برای معادل مبتنی بر لچ [10] ، و احراز شرایط سیگنال ها توسط فاز مناسب ساعت انجام دهیم.

ElastiStore، حداکثر یک نقل مکان را در هر چرخه برای VC آماده می پذیرد. ثبات بقیه VC ها می تواند در طول چرخه های غیر فعال برای صرفه جویی در توان پویا ساعت دار باشد. ثبات اصلی یک VC باید با سنجش قفل شود (الف) هنگامی که آن را از EMPTY به حالت HALF حرکت می کند (در حال پذیرش نقل مکان ورودی جدید است) ، و (ب) هنگامی که از FULL به HALF حرکت می کند (ثبات اصلی توسط محتویات بافر مشترک دوباره پر می شود). در مقابل، بافر به اشتراک گذاشته و منطق کنترل ElastiStore حالت هر VC را همیشه به صورت ساعت دار حفظ می کند. این شکل از ساعت راهگاه، فقط برای نوشتن دو ثبات داده ها، مستقل از تعداد کل VC ها فعال می شود.

TarjomeFa.Com

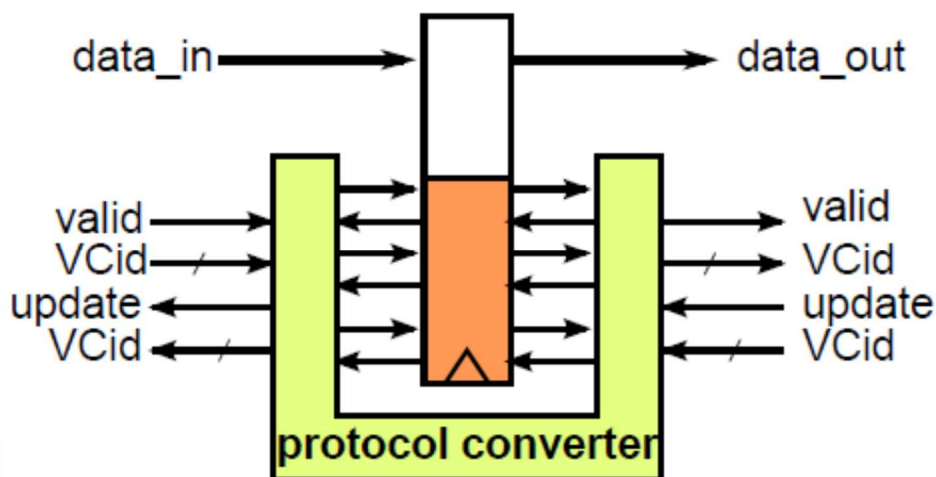
A. تعداد پایین تر سیم های دست دادن

در یک کانال الاستیک، اکثریت سیم ها به سیگنال های داده ها اختصاص می یابند. درصد سیم های دست دادن در کانال های باریک افزایش می یابد است که از بسیاری از VC ها پشتیبانی می کند. حتی اگر منابع سیم کشی فراوان بر روی تراشه برای حمایت جداگانه سیگنال های دست دادن در VC وجود داشته باشد، پروتکل الاستیک هنوز هم می تواند برای به حداقل رساندن تعداد سیم های کنترل اصلاح شود.

اولین تغییر کد V سیگنال های معتبر، که تنها از آن فعال خواهد شد، به یک بیت معتبر در هر کانال و یک VCid شاخص VC انتخاب شده در $\log 2 V$ بیت کدگذاری می کند. اصلاح دوم تعداد سیگنال های آماده را به حداقل می رساند. وقتی VC i ام در گیرنده آماده پذیرش نقل مکان جدید باشد، سیگنال آماده (i) را به طور مستقل از بقیه

VCها ادعا می کند. بنابراین، چند سیگنال آماده را می توان در هر سیکل کلاک تأکید کرد. اگر ما نخواهیم سیم بیشتر را برای فشار برگشتی صرف نماییم - به عنوان مثال، $\log_2 V + 1$ سیم، ما نیاز به ذخیره وضعیت هر یک از VC های پایین دست در فرستنده داریم. در این رویکرد "آماده ذخیره شده"، که به طور موثر یک شکل ابتدایی از کنترل جریان مبتنی بر اعتبار است، فرستنده رد آمادگی هر یک از VC های پایین دست را با چک کردن مقدار از ذخیره بیت های آماده محلی کنترل می کند. بنابراین، گیرنده فقط یک سیگنال به روز رسانی وضعیت و VCid را می فرستد، که مشخص می کند که VC باید وضعیت خود را به آماده، به روز رسانی نماید.

فرستنده، V بیت آماده، یکی برای هر VC پایین دست، و یک بیت برای بافر به اشتراک گذاشته را می فرستد.



شکل 6. پیاده سازی رویکرد آماده ذخیره شده خارج از ElasticStore توسط مبدل های پروتکل انجام می شود.

در این مورد، VC فعال برای ارسال یک نقل مکان جدید واجد شرایط می شو وقتی که هر دو بیت آماده محلی، و یا بیت آماده از بافر به اشتراک گذاشته باشد. برای VC که داده شده است، ما ادعای بیت آماده بافر مشترک را رد می کند. اگر آن را در حال حاضر صفر است، ما ادعایی بیت آماده از بافر به اشتراک گذاشته. هنگامی که یک به روز رسانی برای VC i ام را دریافت، i امین بیت آماده محلی باید تصریح شود. اگر آن را در حال حاضر ادعا می شد، این بروز رسانی به بیت آماده از بافر مشترک اشاره دارد (به عنوان مثال، VC i ام دو نقل مکان های در گذشته فرستاده است، با توجه به غرفه پایین دست، و یکی از آنها را اشغال بافر مشترک). اجرای روش های ذخیره شده،

آماده می تواند در خارج از ElastiStore توسط مبدل های پروتکل انجام می شود ، همانطور که در شکل نشان داده شده است . 6 .

V. انسجام ELASTISTORE در مسیریاب های NOC

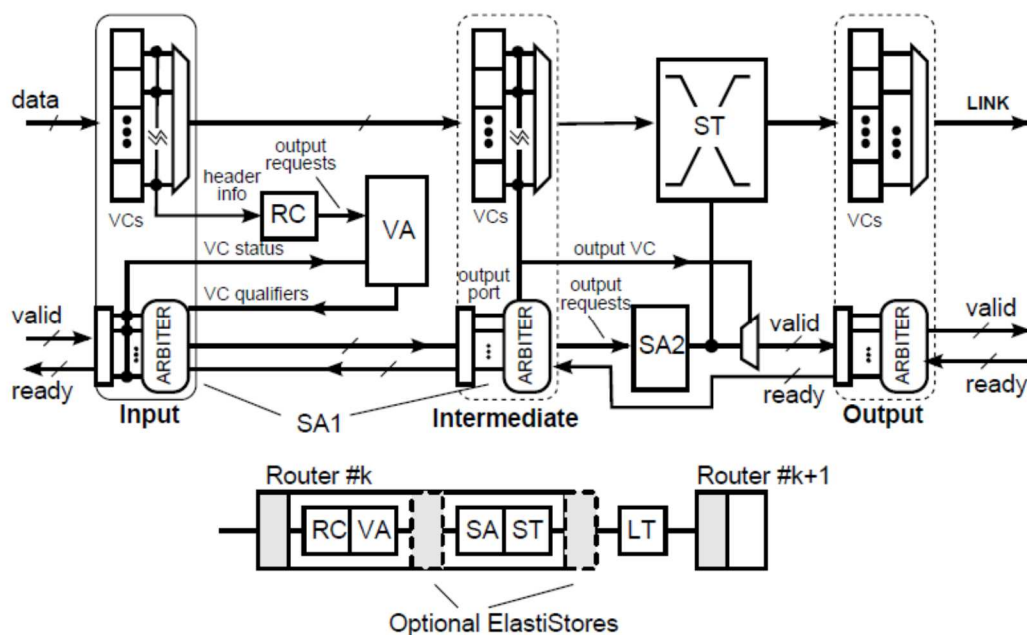
هنگامی که یک بسته وارد یک مسیریاب می شود، به پیدا کردن پورت مقصد خروجی آن از طریق محاسبات مسیریابی (RC) نیاز دارد. پورت خروجی را می توان از پیش در مسیریاب های قبلی با استفاده از (LRC) RC محاسبه نمود. هر بسته باید پس از آن یک VC را در ورودی مسیریاب بعدی ، قبل از خروج از مسیریاب فعلی (شناخته شده به عنوان " خروجی " VC) انتخاب نماید. تطبیق VC های ورودی با VC های خروجی توسط تخصیص (VA) VC انجام می شود. مجاز نمودن بسته به تغییر VC هنگامی می تواند به کار گرفته شود که الگوریتم مسیریابی هیچ محدودیت های VC را تحمیل نکند (به عنوان مثال ، مسیریابی XY حتی نیاز به حضور VC ها ندارد). با این حال، اگر الگوریتم مسیریابی و / یا پروتکل های لایه بالاتر (به عنوان مثال ، وابستگی کش) محدودیت های خاص در استفاده از VC ها را اعمال کند، پس از آن در پرواز دلخواه تغییر VC ممنوع است، چرا که آنها ممکن است به بن بست منجر شوند. در صورت وجود محدودیت های VC ،تخصیص VC همه قوانین در تخصیص VC اجرا برای اطمینان از آزادی بن بست است. چنین محدودیت VC متعامد برای بهره برداری از ElastiStore می باشد.

نقل مکان هایی که یک خروجی VC برای دسترسی به پورت خروجی آن به حکمیت . اگر نقل مکان برنده این مرحله شود - به نام تخصیص سوئیچ (SA) و سازمان یافته در مراحل داوری محلی و جهانی ، SA1 - و SA2 آن را از (- ST سوئیچ پیمایش) می گذرد و ، سپس به لینک های خروجی (LT - پیمایش لینک) نسبت به مسیریاب بعدی [12.] حرکت می کند.

ElastiStore را می توان در ورودی ، در خروجی ، و یا در داخل یک مسیریاب همانطور که در شکل 7 نشان داده شده است، یکپارچه نمود. ماژول ElastiStore بین مراحل مسیریاب قرار می گیرد ، در نتیجه به جای ثبات خط لوله معمولی است. مسیریاب NoC تک مرحله حداکثر دو ماژول ElastiStore نیاز دارد. یکی در ورودی و یک

خروجی . به طور مشابه، یک مسیریاب NOC دو مرحله یکی ماژول های اضافی متوسط (ElastiStore در مجموع سه) بین مراحل اول و دوم از مسیریاب نیاز دارد.

ElastiStore ورودی از تمام سیگنال های خروجی اختیاری نشان داده شده در شکل . 5 بهره گیری می نماید: اطلاعات هدر از هر VC به واحد RC که مقصد خروجی را محاسبه می کند منتقل می شود. سپس ، با استفاده از VC اطلاعات وضعیت از ElastiStore ، نقل مکان های از هر امتحان VC برای به دست آوردن یک خروجی VC در قسمت خروجی VC های خروجی فرستاده می شود.



شکل 7 . ElastiStore را می توان در ورودی ، در خروجی ، و یا در داخل یک مسیریاب یکپارچه سازی نمود.

VC خروجی ElastiStore - در صورت وجود - و یا VC های از ورودی ElastiStore مسیریاب بعدی VC های که هنوز خروجی VC را دریافت نکرده اند می توانند ورودی را ترک کنند. در مقابل، VC های ورودی که به یک خروجی VC تطبیق می یابند می توانند به جلو حرکت کنند و برای یک پورت خروجی داوری نمایند. SA می تواند بسته به حضور ElastiStore در همان چرخه و یا در مراحل بعدی انجام شود SA1 توسط داور داخل ElastiStore (هر دو در ورودی و یا در وسط) انجام می شود و SA2 با دنبال حل و فصل مشاجره برای خروجی مشابه در میان ورودی های مختلف است.

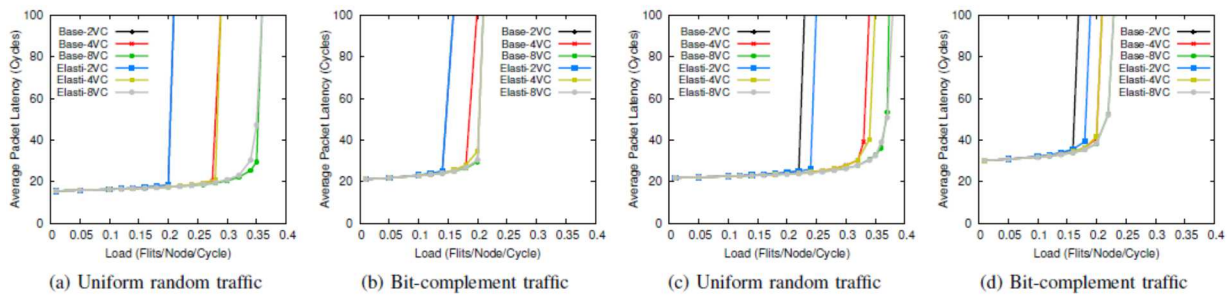
ElastiStore متوسط می تواند نقل مکان هایی را مقرر نماید که خروجی VC را به دست می آورد و رو به جلو حرکت می کند و در انتظار انتخاب توسط SA می ماند. نقل مکان ها باید با دو برچسب ارتقا یابند: پورت مقصد خروجی و خروجی VC از طریق نقل مکان سر بسته به دست می آید. یک بار در داخل ElastiStore متوسط، هر یک از نقل مکان ها به طور مستقل می تواند برای یک پورت خروجی داوری می نمایند. ElastiStore متوسط فقط فرمت ElastiStore ورودی است. نقل مکانی که یک ElastiStore ورودی را ترک می کند و به همان VC حرکت می کند، در ورودی باقی می ماند. یک نقل مکان کردن به خروجی به مقصد خود VC حرکت می کند تنها زمانی که مسیریاب را ترک می کند، و یا زمانی که در ElastiStore خروجی با ادعای سیگنال معتبر مناسب ذخیره می شود. اتصالات آماده / معتبر دست دادن در سراسر مسیریاب، پس از اتصال به شبکه کانال های الاستیک منتقل می شوند. اگر یک مسیریاب با متوسط یا یک خروجی ElastiStore مجهز شود، سیگنال های دست دادن به شیوه ای گام به گام در هر ElastiStore ثبت می شوند. هنگامی که ElastiStores تنها در ورودی قرار داده می شود، سپس ElastiStores متوسط و خروجی به عنوان تغذیه میانی برای داده های آماده / معتبر دست دادن عمل می کنند.

VI بررسی

در این بخش، ما مسیریاب مبتنی بر - ElastiStore را با مسیریابهای مبتنی بر VC معمولی هم از نظر عملکرد شبکه و پیچیدگی سخت افزار مقایسه می نماییم.

A. عملکرد شبکه

تمام مقایسه ها با استفاده از SystemC در شبیه ساز شبکه چرخه-دقیق انجام می شود که تمام اجزای معماری ریز از یک مسیریاب NoC را با فرض $8 \times 8 \times 2$ شبکه های مش با-XY ابعاد دستور مسیریابی مدلسازی می نماید. ارزیابی شامل الگوهای ترافیک مصنوعی با استفاده از یکنواخت ترافیک تصادفی و بیت - مکمل می شود. دیگر الگوهای ترافیک جایگشت از روندهای بسیار شبیه به ترافیک بیتی مکمل پیروی می کند.



شکل 8. منحنی های تاخیر در برابر بار برای مسیریاب های (الف)-(ب) تک مرحله (پ)-(ت) دو مرحله

ترافیک تزریق شامل دو نوع از بسته های اطلاعاتی به تقلید حالات واقعی سیستم می شود : 1 ، نقل مکان بسته های کوتاه (درست مانند بسته های درخواست در یک CMP) ، و بسته های دیگر 5 نقل مکان (درست مانند بسته پاسخ حامل یک خط کش) . برای تجزیه و تحلیل تاخیر توان، فرض کنیم یک توزیع دو پیمانه ای بسته با 50٪ از بسته بودن کوتاه مدت، بسته های 1 ، نقل مکان ، و بقیه که طولانی، بسته های 5 نقل مکان وجود دارد. این درصد با توجه به مطالعات اخیر از ترافیک کش در CMPS در حال اجرای حجم کار در برنامه های واقعی [13] است.

مسیریاب در شروع مطالعه VC را می توان با بافر کم عمق یا عمیق در VC ساخت. با این حال، برای هر یک از VCها، گنجاندن تعداد بافر مورد نیاز برای پوشش اعتباری تاخیر رفت و برگشت مهم است. در یک چرخه مسیریاب پایه ، هر یک از نقل مکان ها در داخل مسیریاب و یک چرخه های اضافی بر روی لینک یک چرخه را صرف می کند. برای جداسازی این لینک از بقیه از مسیریاب، یک بافر خروجی ساده (ثبات خط لوله تک) که در خروجی کراس بار اضافه می شود. در این پیکربندی، هر VC نیاز به 3 بافر برای پوشش دادن زمان رفت و برگشت دارد. این تنظیمات برای دستیابی به عملکرد بالا و نگه داشتن بافر در کل در هر مسیریاب در سطح معقول و منطقی زمانی که تعداد VCها بالا است، کافی می باشد.

جدول 1. مقایسه بافر شروع مطالعه و مسیریاب های مبتنی بر ElastiStore

Stages	Baseline	ElastiStore
1-stage	$3NV + N$	$2(V + 1)N$
2-stage	$4NV + N$	$3(V + 1)N$

به عنوان مثال، یک مسیر یاب پایپ لاین با دو مرحله، اعتبار تاخیر رفت و برگشت را در یک افزایش می دهد - مگر اینکه مسیرهای به روز رسانی اعتباری ترکیبی مستقیم در سراسر مسیر یاب ها به کار گرفته شوند که از مزایای خط لوله محدود است - اما از فرکانس ساعت بالاتر برخوردار است. بنابراین، نیاز به یک حداقل 4 بافر در VC است. تاخیر معادل تک چرخه مسیر یاب مبتنی بر مسیر یاب مبتنی بر VC با یک ElastiStore در ورودی و خروجی است، در حالی که دو چرخه پایپ لاین مسیر یاب معادل مسیر یاب الاستیک است که شامل یک ElastiStore متوسط می باشد.

حتی با این مقدار پایین تر از بافر - مسیر یاب بر اساس - ElastiStore به عملکرد شبکه های مشابه زمانی که به مسیر یاب تک و دو چرخه بر اساس مقایسه VC دستیابی پیدا می کند - که به طور مستقیم به صرفه جویی در مساحت / توان ترجمه می شود. شکل 8 (الف) و (ب) منحنی های تاخیر بار از هر دو مسیر یاب تک چرخه زیر را برای مقایسه زمانی تعداد VC های مختلف تصویر می کشد. در همه موارد، عملکرد مسیر یاب در هر دو بار کم و در بالا غیر قابل تشخیص است . همین نتیجه گیری که با نتایج در شکل 8 (ج) و (د) برای مورد مسیر یاب دو مرحله کشیده شده است. بنابراین، صرفه جویی ElastiStore برای طراح NoC به صورت رایگان، بدون سبک سنگین کردن عملکرد ارائه می گردد.

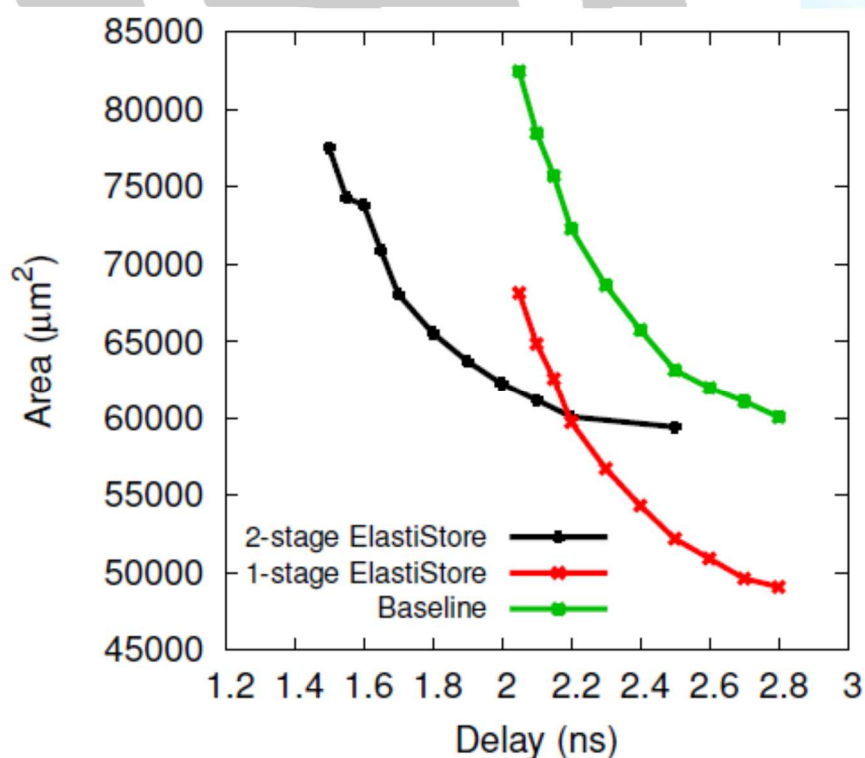
چند شبکه مبتنی بر EB- فیزیکی برای مسیر یابهای ساده [4] ، با نگاشته شد در یک VC ، از سرعت کلاک بالاتر ، با توجه به مرحله حذف VA استفاده می کنند. با این حال، زمانی که مقایسه با مسیر یاب های مبتنی بر ElastiStore تحت پهنای باند دو بخشی شبکه های برابر صورت می گیرد، آنها در عملکرد همانطور که در آزمایش های ما تایید شده است رنج می برند که به دلیل محدودیت فضا، به دلیل تاخیر ترتیب بالای اعمال شده توسط کانال های باریک در شبکه فیزیکی حذف می شوند.

در نهایت، مسیر یاب ها بر اساس VC- را می توان با تخصیص VC مبتنی بر مقصد استاتیک ساده سازی نمود که اجازه نمی دهد که بسته ها VC را تغییر دهند و یکی را با توجه به آنها در منبع نگه می دارد . این ویژگی را می توان برای مسیر یاب ElastiStore استفاده شود. اگر چه این گزینه به نفع پیچیدگی مسیر یاب است، همانطور که در

[14] انجام شده است، تخصیص استاتیک VC (a) توان کلی را با افزایش سر خط مسدود کردن هر شخص VC را کاهش می دهد و (b) مسیریابی تطبیقی را پیچیده می نماید.

B. پیاده سازی سخت افزاری

مسیریاب های ElastiStore تک مرحله و دو مرحله ارائه شده، با استفاده از lookahead RC ، در VHDL اجرا شده است و به کتابخانه استاندارد سلول های صنعتی کم توان 40 نانومتر در شرایط بدترین حالت (0.8 V، 125oC) ، با استفاده از آهنگ جریان اجرای دیجیتال نقشه برداری شده است. مدل های عمومی مسیریاب برای 5 پورت ورودی و خروجی پیکربندی شده است ، همانطور که یک شبکه مش D2 مورد نیاز است و برای VC های در هر پورت، در حالی که عرض نقل مکان در 64 بیت راه اندازی شد. منحنی های مساحت / تاخیر به دست آمده برای تمام طرح ها - پس از ساختن منطق سنتز و ابزار پشتیبانی هدف و استخراج اطلاعات طرح های فیزیکی (هر یک از خروجی با یک سیم از 2 میلی متر لود) - در شکل 9 نشان داده شده است .



شکل 9 نتایج پیاده سازی سخت افزاری مسیریاب های ElastiStore

شکل 9 شامل عملکرد یک چرخه 4 - مسیریاب VC با LRC و 3 بافر در VC ورودی و دو مسیریاب مبتنی بر ElastiStore، با همان تعداد VC ها می شود. هر دو مسیریاب ElastiStore زیر مسیریاب پایه قرار دارند. مسیریاب تک مرحله دارای همان تاخیر مسیریاب مبتنی بر VC است، البته با مساحت به طور قابل توجهی پایین تر، به دلیل کاهش بافر. هر دو مسیریاب تک مرحله می تواند سرعت آن را با استفاده VA-SA [15] بهبود دهد. حتی اگر دو ElastiStore مرحله مسیریاب نیاز به مساحت بیش از مسیریاب تک مرحله ای در نقطه حداقل تاخیر آن داشته باشند، بیشتر مساحت به صرفه زمانی که محدودیت تاخیر آرام شود، به پایان می رسد. در تاخیر بالا، با این حال، بیشتر پیاده سازی ها با مساحت کارآمد مسیریاب ElastiStore تک چرخه است.

مصرف برق، اندازه گیری شده از روند اندازه گیری های مساحت پیروی می کند، با فرض اینکه که هر دو معماری شروع مطالعه و ElastiStore می تواند ساعت دروازه VC های استفاده نشده باشد. راهگامی ساعت، در مورد ما، ساده تر و زیبا تر می باشد. همچنین، از آنجا که یک پیاده سازی بر اساس محکم نگاه داشتن یک مشخصه منحصر به فرد از بافر الاستیک نیست و همچنین می تواند برای طراحی FIFOs بزرگتر استفاده شود، هر دو طرح شروع مطالعه و ElastiStore می توانید آنها را برای کاهش الزامات مساحت / توان آن به تناسب استفاده کند.

VII. کار مرتبط

ElastiStore به دو مسیر تحقیقات قبلی مربوط می شود: (الف) کنترل جریان الاستیک، و (ب) به اشتراک گذاشته بافر در مسیریابهای NOC:

بافرهای الاستیک در [4] به عنوان یک مکانیسم بافر کم هزینه استفاده می شوند که جایگزین بافر یکپارچه از مسیریاب کرم چاله ی می شود. پشتیبانی از VC های از طریق ترکیبی کنترل جریان EB / VC که از ترکیبی از EBS (در لینک) و بافر منظم VC از طریق چند شبکه فیزیکی ، و یا [16]، [6] ارائه شد. تکنیک های مشابه هیبرید در [8]، [7] ، با ویژگی های اضافی استفاده شده است که فضای بافر VC همراه EB های میان VC های به اشتراک گذاشته شده است. در حالی که این راه حل های موفقیت برای جدایی از ترافیک جریان را میسر می سازد (با بهره گیری از بافر VC)، آنها هنوز هم روی رابط کاربری واحد معتبر / آماده بین EBS در هر یک از لینک تکیه می

کنند. این رابط می تواند جریان های مختلف را تشخیص دهد. بنابراین، چالش این است که اطمینان حاصل شود که تمام نقل مکان ها از VC مسدود شده از EBS در مد منظم صورت گیرد (به عنوان مثال، در یک بافر VC با ترک در قرار داده شده)، به طوری که EBS در لینک بتواند برای باز کردن VC مورد استفاده قرار گیرد. برای رسیدن به این هدف، معماری EB / VC هیبرید در منطق کنترل غیر بدیهی است که مختصات جریان ترافیک VC تکیه می کنند. در عوض، ElastiStore از این عارضه با استفاده از رابط های دست دادن فرد (و یا سیاست های ذخیره شده) برای هر یک از حمایت های VC جلوگیری می نماید به طوری که VC جریان های مختلف و ترافیک ذاتا منطقی از هم جدا می شوند و به راحتی به حافظه مربوطه بافر موازی آن هدایت می شوند. حضور یک بافر مشترک در ElastiStore ابزاری در بهینه سازی استفاده از فضای بافر موجود در مسیریاب NoC است. در اصل، مکانیسم پیشنهادی رسیدن برای همان هدف به طور قابل توجهی برای بافر به اشتراک گذاشته و طرح های بافر دیگر [17]، [18]، [19]، [20] گران تر است.

در طرح های مشترک، بافر، طراحان عمدتا از هر دو لیست های پیوندی [18]، و یا روش های مبتنی بر جدول [17] به منظور هماهنگ کردن جریان ترافیک از طریق بافر استفاده می نمایند. هر VC باید اشاره گر خود را برای شناسایی نقل مکان های آن در بافر، صرف نظر از اندازه بافر تنظیم نماید، در حالی که از بن بست اجتناب می شود. در حالی که هزینه های منطق کنترل در مسیریاب با فضای بافر بزرگ مستهلک می شود، سربار در مسیریابهای کم هزینه با فضای بافر ElastiStore به طور قابل توجهی حداقل می شود. نیاز به هیچ منطق اشاره گر و جلوگیری از بن بستها در سطح پروتکل های ساخت و ساز نیست که در نتیجه ارائه راه حل بی هزینه برای این مسئله حیاتی است.

VIII. نتیجه گیری

همانطور که سیستم های چند هسته ای به عرصه سیستم هایی با هسته های بسیار زیاد گذار می نمایند، فشار بر شبکه اتصال به طور قابل ملاحظه ای افزایش می یابد. انتظار می رود NOC خواسته های در حال گسترش برای تعداد روزافزون از عناصر پردازش را انجام دهد در حالی که - در همان زمان - رد پای مساحت / توان به شدت

محدود شده باقی می ماند. از این رو، طرح های کم هزینه NOC که برای استفاده و عمل با زمان تاخیر کم برای مقیاس پذیری مورد نیاز می باشند، در آینده ضروری است. بافر NOC مسیریاب، مصرف کنندگان عمده مساحت و توان و توانمند کلیدی با کارایی بالا می باشند. علاوه بر این، بافر به منظور تسهیل VCها استفاده می شود که ابزاری در افزایش عملکرد بیشتر است و آزادی بن بست ها را میسر می سازد. در این مقاله، ما ElastiStore را پیشنهاد نمودیم که مفهوم بافر الاستیک را به VCهای متعدد گسترش می دهد. ElastiStore الزامات بافر را نزدیک به حداقل ممکن را کاهش می دهد، در حالی که هنوز هم به عملکرد مشابه بسیار گران تر، مسیریاب VC بر اساس سنتی دستیابی پیدا می کند.

REFERENCES

- [1] J. Handy, "NoC interconnect improves SoC economics," *Objective analysis - Semiconductor market research*, 2011.
- [2] W. J. Dally, "Virtual-Channel Flow Control," in *Proc. of the Intl. Symp. on Computer Architecture*, May 1990, pp. 60-68.
- [3] J. Browne, "On-Chip Communications Network Report, 2012."
- [4] G. Micheliogiannakis, J. Balfour, and W. J. Dally, "Elastic buffer flow control for on-chip networks," in *IEEE Int. Symp. on High Performance Computer Architecture*, 2009.
- [5] A. Roca, C. Hernandez, J. Flich, F. Silla, and J. Duato, "Silicon-aware distributed switch architecture for on-chip networks," *Journal of Systems Architecture*, vol. 59, no. 7, pp. 505 - 515, 2013.
- [6] G. Micheliogiannakis and W. Dally, "Elastic buffer flow control for on-chip networks," *IEEE Trans. on Computers*, vol. 62, no. 2, Feb. 2013.
- [7] S. M. Hassan and S. Yalamanchili, "Centralized buffer router: A low latency, low power router for high radix noCs," in *IEEE/ACM International Symposium on Network on Chip*, April 2013.
- [8] A. K. Kodi, A. Sarathy, and A. Louri, "ideal: Inter-router dual-function energy and area-efficient links for network-on-chip (noc) architectures," in *Proc. of Intl Symp. on Comp. Architecture*, 2008, pp. 241-250.
- [9] M. M. K. Martin, D. J. Sorin, B. M. Beckmann, M. R. Marty, M. Xu, A. R. Alameldeen, K. E. Moore, M. D. Hill, and D. A. Wood, "Multifacet's general execution-driven multiprocessor simulator (gems) toolset," *SIGARCH Computer Architecture News*, vol. 33, 2005.
- [10] J. Cortadella, M. Kishinevsky, and B. Grundmann, "Synthesis of Synchronous Elastic Architectures," in *Proc. ACM/IEEE Design Automation Conference*, Jul. 2006, pp. 657-662.
- [11] OCP-IP protocol specification. www.ocp-ip.org
- [12] W. J. Dally and B. Towles, *Principles and Practices of Interconnection Networks*. Morgan Kaufmann, 2004.
- [13] S. Ma, N. Enright Jerger, and Z. Wang, "Whole Packet Forwarding: Efficient Design of Fully Adaptive Routing Algorithms for Networks-on-Chip," in *Proc. of the Intern. Symp. on High Performance Computer Architecture*, Feb. 2012, pp. 467-478.
- [14] F. Gilibert and et al., "Improved utilization of noc channel bandwidth by switch replication for cost-effective multi-processor systems-on-chip," in *NOCS*, 2010, pp. 165-172.
- [15] R. D. Mullins, A. F. West, and S. W. Moore, "Low-latency virtual-channel routers for on-chip networks," in *Proc. of the Intl. Symp. on Computer Architecture*, 2004, pp. 188-197.
- [16] B. Grot, J. Hestness, S. W. Keckler, and O. Mutlu, "A QoS-Enabled On-Die Interconnect Fabric for Kilo-Node Chips," *IEEE Micro*, vol. 32, no. 3, May 2012.
- [17] C. Nicopoulos and et al., "Vichar: A dynamic virtual channel regulator for network-on-chip routers," in *IEEE/ACM Intern. Symp. on Microarchitecture*, 2006, pp. 333-346.
- [18] M. Lai, Z. Wang, L. Gao, H. Lu, and K. Dai, "A Dynamically-Allocated Virtual Channel Architecture with Congestion Awareness for On-Chip Routers," in *Design Automation Conference*, 2008.
- [19] W. Su, J. S. Shen, and P. A. Hsiung, "Network-on-Chip Router Design with Buffer-Stealing," in *ASP-Design Automation Conference*, 2011.
- [20] A. T. Tran and B. M. Baas, "RoShaQ: High-performance on-chip router with shared queues," in *IEEE Intern. Conf. on Computer Design*, Oct. 2011, pp. 232-238.

برای خرید فرمت ورد این ترجمه، بدون واتر مارک، اینجا کلیک نمایید.



این مقاله، از سری مقالات ترجمه شده رایگان سایت ترجمه فا میباشد که با فرمت PDF در اختیار شما عزیزان قرار گرفته است. در صورت تمایل میتوانید با کلیک بر روی دکمه های زیر از سایر مقالات نیز استفاده نمایید:

لیست مقالات ترجمه شده ✓

لیست مقالات ترجمه شده رایگان ✓

لیست جدیدترین مقالات انگلیسی ISI ✓

سایت ترجمه فا ؛ مرجع جدیدترین مقالات ترجمه شده از نشریات معتبر خارجی