**عنوان پژوهش انجام­شده**

---1، محسن انصاری۲، علیرضا اجلالی۳

۱ دانشجوي کارشناسی ارشد، دانشکده‌ي مهندسي کامپيوتر، دانشگاه صنعتي شريف

تهران، ايران

[----@ce.sharif.edu](mailto:----@ce.sharif.edu)

۲ دانشجوي دکتری، دانشکده‌ي مهندسي کامپيوتر، دانشگاه صنعتي شريف

تهران، ايران

[mansari@ce.sharif.edu](mailto:mansari@ce.sharif.edu),

۳ دانشیار، دانشکده‌ي مهندسي کامپيوتر، دانشگاه صنعتي شريف

تهران، ايران

hessabi@sharif.edu

**چکيده**

امروزه با توجه به آن­که استفاده از اتصالات الکتریکی در شبکه بر تراشه منجر به بروز مشکلات عمده­ای از جمله افزایش تأخیر بخش ارتباطی، توان مصرفی و مساحت اشغالی در سیستم­های با کارآیی بالا شده است؛ رفع اشکالات این حوزه بسیار با اهمیت است. یکی از اشکالات متداول در این شبکه­ها همشنوایی است. برای مقابله با همشنوایی از انواع کدهای اجتناب از همشنوایی در کدگذار-کدگشاهای تعبیه­شده در شبکه بر روی تراشه برای ارسال داده درون تراشه استفاده می­شود. زیرا که با پیشرفت فناوری معماری­های شبکه بر روی تراشه، سربار تحمیلی این­گونه کدگذار-کدگشاها از نظر مساحت، تأخیر وتوان بسیار مهم ارزیابی شده است، تأخیر انتشار در سرتاسرگذرگاه­های روی تراشه ناشی از انتقال داده روی سیم­ها در جهات مختلف است. با روی کارآمدن روش سیستم اعداد فیبوناچی، تحولی در کدینگ شبکه بر روی تراشه به جهت اجتناب از همشنوایی به­وجود آمد. بنابراین طرح تکنیک­های جدید در رابطه با کدینگ داده­های ارسالی در شبکه بر تراشه حائز اهمیت است. همچنین سیستم عددی مبتنی بر فیبوناچی که از مجموعه­ی دیجیتال 0 و 1 استفاده می­کند دارای ویژگی­های بسیار قابل توجهی است که برای انواع کاربردها می­توان از آن ایده گرفت. بنابراین استفاده­ی کاربردی از آن در شبکه بر تراشه نیز می­تواند سودمند باشد. بنابراین برای کدینگ ارسال داده در اتصالات شبکه بر تراشه می­توان از این سیستم عددی استفاده کرد.

کلمات کليدي

شبکه بر تراشه، همشنوایی، کدهای اجتناب از همشنوایی، کدهای مبتنی بر سیستم اعداد فیبوناچی

# مقدمه

پیشرفت­های اخیر در زمینه­ی فناوریِ طراحی مدارهای خیلی مجتمع[[1]](#footnote-1)، طراحان را قادر ساخته است تا واحدهای پیچیده­ای مانند هسته­های پردازشی، واحدهای حافظه، واحدهای ورودی/خروجی و غیره را در قالب یک سامانه[[2]](#footnote-2)، بر روی یک تراشه[[3]](#footnote-3) پیاده­سازی کنند. وجود چندین واحد پردازشی روی یک تراشه، سامانه­ی روی تراشه[[4]](#footnote-4) را به سخت­افزارهای پیچیده­ای با قدرت پردازشی بالا مبدل می­سازد [1].

با پيشرفت فناوری و كوچكترشدن روزافزون اندازه ترانزيستورها، پيچيدگي سيستم­هاي چندپردازنده­اي بر روي تراشه­ها در حال افزايش است. افزايش تعداد منابع پردازشي موجود در طراحي تراشه­ نيازمند استفاده از يك بستر ارتباطي مناسب براي برقراري اتصالات ميان آن­ها مي­باشد. به طور كلي ساختارهاي ارتباطي بر روي تراشه­ها بايستي شامل ويژگي­هاي اساسي از جمله كارآيي بالا، ساخت­يافته بودن، قابليت استفاده­ی مجدد و مقيا­س­پذيري باشند. در ابتدا سيستم بر روي تراشه­ها با بكارگيري ساختار گذرگاه، به عنوان يك تكنيك پياده­سازي براي حل اين مشكل معرفي شدند. ساختار گذرگاه كه در سيستم بر روي تراشه­ها مورد استفاده قرار گرفته است، ويژگي مقياس­پذيري را ندارد و در مورد قابليت استفاده مجدد نيز محدوديت­هايي دارد. به همين علت مفهوم جديدي تحت عنوان شبكه برروي تراشه­ها براي حل اين مشكل معرفي گرديد [3].

اگرچه شبکه بر تراشه در ابتدا با هدف بهبود سرعت انتقال مطرح شد، ولی با توجه به اهمیت کنونی توان مصرفی، بررسی عوامل مصرف توان در شبکه بر روی تراشه و کنترل آن­ها امری اجتناب­ناپذیر است. امروزه بخش قابل توجهی از توان مصرفی در یک سیستم روی تراشه متعلق به شبکه بر روی تراشه­ی موجود در آن است.

## همشنوایی و چالش­های اساسی آن

يكي از پديده­هايي كه در مسيرياب­هاي بر روي شبكه به عنوان يك اشكال شناخته مي­شود، اثر ناشي از اشکال همشنوايي در لينك­هاي ارتباطي است. در اين پديده، يك سيگنال در حال انتقال در يك مدار و يا از طريق يك كانال ارتباطي، بر روي مدار يا كانال ارتباطي ديگر موجود در سيستم تأثير نامطلوبي مي­گذارد. در واقع نويز همشنوايي از جفت­شدگي انرژي الكتريكي انتقالي میان رساناهاي الكتريكي ناشي مي­شود.

سه نوع اصلي اين جفت­شدگي­ها خازني، رسانايي و عايقي می­باشد. جفت­شدگي­ خازني زماني ناشي مي­شود كه دو رساناي مجزا در فاصله­­ی نزديكي از هم قرار گرفته و به همچون يك خازن رفتار كنند. جفت­شدگي عايقي زماني رخ مي­دهد كه جريان الكتريكي موجود در يك رسانا منجر به برقراري جريان مشابه در رساناي ديگر گردد. جفت­شدگيِ رسانايي نيز زماني رخ مي­دهد كه يك اتصال فيزيكي بين رساناهاي مورد نظر برقرار باشد.

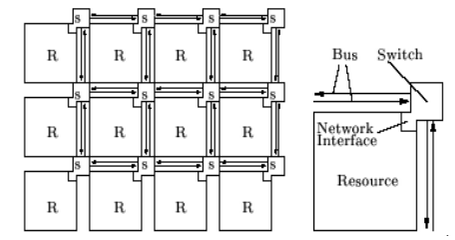
همشنوایی در میان سیم­های همسایه یکی از چالش­های اساسی در ارتباطات بر روی تراشه­ها می­باشد. همشنوایی می­تواند باعث از بین رفتن هم­راستاییِ زمانی در انتقال داده­ها و همچنین باعث مصرف توان پویایِ بالایی گردد. از آن­جا که همشنوایی وابسته به طول مسیرهای ارتباطی، الگوی داده­های ارسالی خط، مباحث فیزیکی و غیره می­باشد، انواع کدینگ­ها جهت مقابله با اشکالات همشنوایی مطرح شده­است.

## مفاهیم بنیادی

شبکه­های روی تراشه به عنوان زیرساختی برای ارتباط هسته­های پردازشی موجود در یک تراشه ارائه شدند. به طور کلی این نوع شبکه­ها از سه جز اصلی تشکیل می­شوند که عبارتند از:

* منابع پردازشی یا هسته­های پردازشی
* راه­گزین­ها[[5]](#footnote-5) و مسیریاب­ها[[6]](#footnote-6)
* اتصالات[[7]](#footnote-7) یا کانال­های ارتباطی

شکل ‏1‑1 ساختار کلی یک شبکه بر روی تراشه را نشان می­دهد که در آن منابع پردازشی با R و راه­گزین­ها با S نشان داده شده­اند. در ادامه به توصیف هر یک از اجزای شبکه بر تراشه پرداخته می­شود [2].

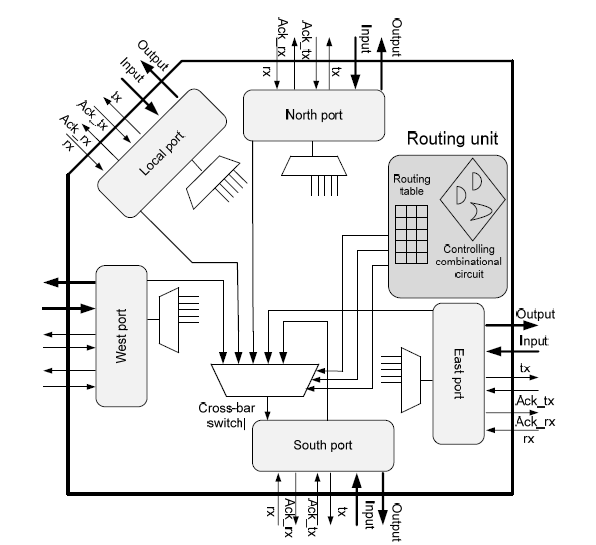


شکل ‏1‑1: نمایشی از اجزای شبکه بر تراشه [2]

### عناصر مسیریاب

يكي از مهمترين و اصلي­ترين بخش­هاي يك مسيرياب را واحد مسيريابي آن تشكيل مي­دهد كه وظيفه­ی پياده­سازي الگوريتم مسيريابي را برعهده دارد. در واقع اين واحد با دريافت يك واحد سرآيند توسط بافر ورودي و اطلاع از وجود چنين فیليتي[[8]](#footnote-8)، فعال شده و با انجام يك سري مقايسه بين آدرس مسيرياب مربوطه در شبكه و همچنين آدرس مقصد كه در فیليت سرآيند موجود است، پورت خروجي مناسبي را براي بست­هایی كه در سرآیند آن به بافر ورودی آمده است را انجام می­دهد. همانگونه که در شکل ‏1‑2 نشان داده شده­است این واحد نيز از بخش­هاي مختلفي تشكيل شده كه هر يك وظايف مربوط به خود را انجام مي­دهند. فرآيند و عملكرد اين واحد در واقع به اين گونه است كه با دريافت يك فیليت سرآيند ابتدا اطلاعات مربوط به مسيريابي از آن را جدا نموده و پس از بررسي­هاي مربوط به الگوريتم مسيريابي XY یک پورت را برای آن انتخاب مي­كند. اما يك واحد مهم به نام واحد داوري در اين­جا وجود دارد كه تخصيص عادلانه پورت­هاي خروجي به هر يك از پورت­هاي ورودي درخواست كننده پورت خروجي موردنظر را عملي مي­كند. در واقع واحد داوري زماني فعال مي­گردد كه بيش از يك درخواست براي دسترسي به يك پورت خارجي وجود داشته باشد. اين واحد با دريافت تقاضاهاي مختلف براي يك پورت خارجي خاص نام پورت­هاي ورودي درخواست كننده را با همان ترتيبي كه آمده­اند در يك صف ذخيره نموده و پس از آزاد شدن پورت خروجي موردنظر پيش از پاسخ­دهي به بسته­هاي جديد وارد شده صف انتظار مربوط به هر يك از پورت­هاي خروجي را بررسي مي نمايد.

در صورت وجود درخواستي در صف مربوطه، ابتدا به آن­ها پاسخدهي مي­نماید. در اين حالت بسته­ی ورودي جديد نيز به علت اشغا­ل­شدن پورت خروجي مربوطه در صف انتظار براي دستيابي به آن قرار مي­گيرد. اهميت وجود واحد داوري زماني كه مسيرياب شبيه­سازي شده را در شبكه با همبندي مفروضي قرار مي­دهيم قابل لمس است، چرا كه در صورت عدم استفاده از واحد داوري احتمال وقوع پديده­ی گرسنگي[[9]](#footnote-9) را براي يك سري از بسته­ها در پی خواهد داشت.



شکل ‏1‑2: ساختار کلی یک مسیریاب شبکه بر روی تراشه [2]

به اين ترتیب كه آن­ها در انتظار نامتناهي براي دريافت يك پورت خروجي باقي مي­مانند. با توجه به سوييچينگ خزشی[[10]](#footnote-10) كه در آن يك مسير ارتباطي تا ارسال كامل بسته به آن اختصاص مي­يابد و همچنين اين نكته كه سايز بافرها در اين سوييچينگ از طول بسته نيز بايستي كمتر باشد، وقوع پديده­ی گرسنگي براي يك بسته علاوه بر مشكلاتي كه براي آن بسته به وجود مي­آورد، يك­سري از بافرها و پورت­هاي خروجي درون شبكه را نيز اشغال نموده و به عبارت ساده­تر منجر به وقوع ازدحام در شبكه مي­گردد.

### عناصر ارتباطی

اتصالات سیم­هایی هستند که برای اتصال بلوک­های عملیاتی به یکدیگر استفاده می­شوند. برای حفظ بازده خروجی مناسب، در انتخاب طول سیم باید دقت نمود ویا از یک طرح خط لوله و یا کدگذاری داده به منظور مقابله با تأخیر در آن­ها استفاده کرد.

## اهمیت تحمل­پذیری اشکال در شبکه بر روی تراشه

استفاده از فناوري نانوالكترونيك از جنبه­هاي ديگر نيز بر روي طراحي سيستم­هاي محاسباتي اثر مي­گذارد. كوچک­ترشدن ظرفيت خازن­هاي مدار و كاهش سطوح ولتاژ منبع تغذيه و منطق، حساسيت گيت­ها، فليپ­فلاپ­ها و واحدهاي حافظه مورد استفاده را در برابر انواع نويزهاي محيطي، ذرات باردار و نويزهاي همشنوايي را كه مي­توانند منجر به ايجاد خطاهاي گذرا و دايمي شوند را افزايش داده است.

به همین دلیل طراحی سیستم­های تحمل­پذیر اشکال که بتوانند مانع از تولید نتایج نادرست یا آثار مخرب شوند و یا با استفاده از روش­هایی بتوانند خود را از وضعیت خطای ایجادشده در سیستم یا شکست کامل سیستم بازیابی نمایند، اهمیت ویژه­ای یافته است.

پیچیده­ترشدن طراحی­ها و امکان استفاده از تعداد بیشتری ترانزیستور بر روی یک تراشه از دو جنبه­ی دیگر نیز حائز اهمیت است. یکی این­که هزینه­های طراحی، ساخت، تأیید صحت و آزمون تراشه­های مذکور افزایش می­یابد. با توجه به عدم عملکرد کاملاً صحیح مولفه­­ها و ساختارهای ارتباطی در هنگام ساخت، اضافه­کردن مشخصه­ی تحمل­پذیری اشکال به صورت جزئی از طراحی که بر روی تراشه­ها پیاده­سازی می­شود، مورد توجه قرار می­گیرد. جنبه­ی دیگر این است که پیچیده­ترشدن طراحی باعث افزایش احتمال رخداد خطاهای انسانی هنگام طراحی و پیاده­سازی می­شود که تشخیص و برطرف­کردن آن­ها به کمک روش­های تأیید صحت به هزینه­ی بیشتری احتیاج دارد.

با توجه به موارد بیان­شده، طراحی شبکه بر تراشه بر روی تراشه­های قابل اعتماد که هم از روش­های بهتری برای افزایش اطمینان، احتمال بازیابی از شکست و تجمل­پذیری اشکال استفاده می­کنند و هم خطاهای انسانی را بهتر تشخیص داده و برطرف می­کنند، اهمیت به­سزایی دارد.

الگوریتم­های مسیریابی تحمل­پذیر اشکال بیشتر برای مقاوم­کردن شبکه­ی ارتباطی در برابر اشکال­های دایمی طراحی می­شوند. در بیشتر کارهای انجام شده الگوریتم­های مسیریابی برای وضعیتی که یک یا چند ناحیه از شبکه به علت وقوع اشکال دایمی دچار مشکل شده است، طراحی شده­اند.

## فعالیت­های پیش­رو

برای تأمین پروژه­ی مربوطه ابتدا بایستی روش­های کدینگ مورد استفاده در مقابله با اشکالات همشنوایی مورد مطالعه و بررسی قرار داده­شود و سپس به ارزیابی کلاس­های مختلف اشکالات همشنوایی از جمله 0C، 1C، 2C، 3C، 4C پرداخته و در پایان کدینگ­های مختلف را از نظر سربار، قابلیت اطمینان، زمان­بندی و مصرف توان با هم مقایسه نمود.

# کدهای اجتناب از همشنوایی

از ميان مسائل چالش برانگيز، قابليت اطمينان نقش كليدي را در طراحي و پياده سازي شبکه بر روي تراشه دارد. همشنوایی از مهم­ترین اشکالات گذرای سیستم است که قابلیت اطمینان هر سیستم روی تراشه­ای را دچار تهدید می­کند. نتایج تحقیقاتی این­گونه بیان می­دارند که به دلیل ماهیت نزدیکی سیم­های شبکه بر تراشه، اشکال همشنوایی از درصد بالاتری برای اشکال در ماهیت داده­های ارسالی ایفا می­کند. از طرفی سیم­های موجود در شبکه بر تراشه به صورت موازی و در طول­های بالا تعبیه می­شوند. این سیم­های موازی و طویل باعث بوجود­آمدن خازن­های جفت­شدگی بین سیم­های مجاور می­شوند که به نوبه­ی خود اشکال همشنوایی را روی برخی از سیم­های کانال­های شبکه بر روی تراشه ایجاد می­کنند.

چندین روش برای مقابله با اشکالات همشنوایی ارائه شده است. در سطح طراحی، ایده­هایی برای اصلاح طراحی تراشه و سعی در کاهش خازن­های جفت­شده پیشنهاد شده است. روش­های مختلف مسیریابی روی تراشه از این ایده­ها می­باشد. یکی از ساده­ترین روش­ها برای کاهش اثر همشنوایی روش محافظ­گذاری[[11]](#footnote-11) برای سیم­هاست. در محافظ­گذاری فعال[[12]](#footnote-12) با دو برابرکردن تعداد سیم­ها می­توان تأخیر گذرگاه را تا حدود نصف کاهش داد. اما در این روش با افزایش 100 درصدی مساحت روبرو هستیم. در محافظ­گذاری غیر فعال[[13]](#footnote-13) نیز می­توان با صرف مساحت دوبرابری تأخیر گذرگاه را تا 75درصد کاهش داد. سربار بیش از حد این دو روش باعث شده است تا طراحان و محققان به سمت استفاده از این دو روش نروند.

## کدهای گذار ممنوعه

کدهای اجتناب از همشنوایی برای رفع اشکالات همشنوایی در انواع کانال­های شبکه بر تراشه به­وجود آمده­اند. کدهای گذار ممنوعه برخی از گذارهای نامعتبر را حذف می­کنند تا سرعت کانال­ها و صحت ارسال داده­های ارسالی افزایش یابد. اما در عین حال کدگذار-کدگشاهایی که بر این اساس کار می­کنند با سربار مساحت، تأخیر و توان مصرفی همراه هستند. پیچیدگی کدگذار-کدگشاهای مبتنی بر کدهای گذار ممنوعه می­تواند با تقسیم­بندی کانال­ها رفع گردد. در این روش­ها کانال­ها به چندین کانال با عرض کمتر تقسیم می­شوند و کدگذاری به صورت جداگانه برای هر کانالِ تقسیم­شده انجام می­گیرد. همچنین برای جلوگیری از الگوهای نامعتبر در کانال­ها می­توان از روش محافظ­گذاری استفاده کرد که باعث افزایش هزینه، مساحت و توان مصرفی خواهد شد.

از آن­جا که رخداد گذارهای کلاس­های 3C و 4C در کانال­های شبکه به دلیل وجود الگوهای بیتی 010 و 101 در فلیت­های اطلاعات است، یا کاهش تعداد رخدادِ این الگوهای بیتی به مقابله با اشکال همشنوایی پرداخت. در کدگذاری الگوی ممنوعه از رخداد این الگوهای بیتی در فلیت­ها ممانعت می­شود. یک کلمه­ی کد، عاری از الگوی ممنوعه نامیده می­شود هرگاه هیچ الگوی بیتی ممنوعه­ای در هیچ­یک از سه بیت متوالی آن مشاهده نشود. به عنوان مثال 1011101 یک کلمه­ی کد عاری از الگوی ممنوعه نیست زیراکه دارای زیر الگوی 101 است؛ در حالی که 1100110 یک کلمه­ی کد عاری از الگوی ممنوعه است. اگر در یک سیستم کدگذاری، کلمه­های کد تماماً عاری از الگوهای ممنوعه باشند، تأخیر بدترین حالت مشاهده شده بر روی کانال بیشتر از 2C نخواهد بود. بنابراین با کدگذاری کلمه­های داده به کلمه­های کد عاری از الگوهای ممنوعه می­توان به تسریعی تقریباً 100 درصد دست یافت [3].

در کدگذاری عاری از الگوهای ممنوعه نیاز است که یک الگوریتم نگاشت، کلمه­ی کد ورودی را به یک کلمه­ی کد عاری از الگوی ممنوعه نگاشت کند. در کتابچه­ی کدی که تعداد کد عاری از الگوی ممنوعه­ی آن­ها توانی از دو نیست، الگوریتم نگاشت مذکور یک به یک نخواهد بود. در این شرایط می­توان با انجام بهینه­سازی در الگوریتم نگاشت پیچیدگی کدگذار و کدگشا را کاهش داد. اما با افزایش عرض کانال ارتباطی، پیچیدگی کدگذار و کدگشا به صورت نمایی زیاد می­شود و همچنین تعداد کل جایگشت­های قابل اعمال به سرعت، رشد می­کند. به عنون مثال برای یک کدگذار کوچک 4 به 5 اجتناب از همشنوایی 1012×2 جایگشت نگاشت می­توان تعریف کرد [3].

کدهای الگو ممنوعه الگوهای نامعتبر 101 و 010 را حذف کرده تا الگوهای گذار نامعتبر را از کانال­ها از بین ببرند. کدهای الگو ممنوعه قابلیت بهینه­شدن به جهت کاهش سربار احتمالی سرعت، مصرف توان و مساحت کدگذار-کدگشاهای شبکه بر تراشه را دارند. اما از طرفی دیگر نیز در کانال­های با عرض زیاد استفاده از سیستم­های عددی برای کاهش سربارهای کدگذار-کدگشاهای موجود در تراشه پیشنهاد شده است. شرایط قابل قبول برای استفاده از سیستم عددی در کدگذار-کدگشاهای موجود در زیر آورده شده است:

*سیستم عددی باید کامل باشد بدین معنا که هر عدد حداقل یک نمایش در سیستم عددی داشته باشد. سیستم عددی باید بتواند هر عدد موجود را به یک کد الگو ممنوعه تبدیل کند. به طور کلی، حداکثر تعداد کدهای الگو ممنوعه در یک فضای* K *بیتی برابر با است که* K+1 *امین عنصر دنباله­ی فیبوناچی است [3].*

برای این منظور الگوهای مجاز در این سیستم در جدول ‏**2‑1** برای کدهای الگو ممنوعه­ی سه وچهار بیتی به همراه سربارشان با توجه به فضای تعداد بیت­ها مشخص کرده است. همانطور که در جدول نمایش داده شده است الگوهای نامعتبر در کانال­های سه بیتی برابر با دو حالت و در چهاربیتی شش حالت است که به ترتیب 25 درصد و 5.37 درصد از کل حالات را شامل می­شود. با توجه به روند صعودی تعداد بیت­های هر کانال این درصد الگوهای نامعتبر نیز افزایش می­یابد؛ به طوری که در کانال­های 5 بیتی برابر با 50 درصد می­شود و در کانال­های 10بیتی حدود 81 درصد خواهد بود.

**جدول ‏2‑1: حالات معتبر و نامعتبر کدهای الگو ممنوعه­ی سه و چهار بیتی**

|  |  |  |
| --- | --- | --- |
| **حالات نامعتبر** | **مجموعه حالات معتبر کدهای الگو ممنوعه** | **عرض کانال** |
| **101،010** | **000، 001، 011، 100، 110، 111** | **3** |
| **1010، 0101، 1011، 1101، 0010، 0100** | **0000، 0001، 0011، 0110، 0111، 1000، 1001، 1100، 1110، 1111** | **4** |

### سیستم عددی دودویی بر پایه­ی فیبوناچی

روش کدگذاری الگو ممنوعه بر مبنای سیستم عددی فیبوناچی، از دنباله اعداد فیبوناچی به عنوان پایه­های خود استفاده می­کند؛ به عنوان مثال برای فضای چهاربیتی از دنباله­ی اعداد 1 1 2 3، در فضای پنج بیتی از دنباله اعداد 1 1 2 3 5 و در فضای شش بیتی 1 1 2 3 5 8 استفاده می­کند. ایده­ی اصلی روش کدگذاری مبتنی بر فیبوناچی این است که هر الگوی بیتی 011 با 100 قابل جابه­جایی است. بدین معنا که می­توان الگوی 011 را با 100 عوض نمود. بر این اساس برخی الگوهای نامعتبر حذف می­شوند.

یک سیستم عددی یک چارچوب است که در آن اعداد توسط ارقام نشان داده می­شوند. متداول­ترین سیستم عددی در طراحی دیجیتال سیستم عددی دودویی است که از توان­های دو به عنوان پایه استفاده می­کند. نمایش دودویی یک عدد در فرمول 1 نشان داده شده است. سیستم عددی دودویی کامل و غیر مبهم است، به این معنا که هر عدد، یک و تنها یک نمایش در سیستم عددی دارد.

سیستم عددی مبتنی بر پایه­ی فیبوناچی یک سیستم عددی است که از دنباله­های فیبوناچی به عنوان پایه استفاده می­کند. نمایش فیبوناچی یک عدد در فرمول 2 نشان داده شده است. همچنین تعریف تابع فیبوناچی در فرمول 3 قرار داده شده است.

## کد­های غیرهمپوشان

در کدهای غیرهمپوشان هیچ دو سیم همسایه­ای در هنگام عبور از یک کلمه­ی کد به کلمه­ی دیگر مجاز نیستند گذاری در خلاف جهت یکدیگر داشته باشند. به عبارت دیگر از الگوهای گذار 10 به 01 و 01 به 10 در این­گونه رمزها اجتناب می­شود. فرض کنید که بین دو سیم همسایه یک کرانه وجود داشته باشد. اگر در تمامی کلمه­های کد بین این دو کد فقط کلمه­های کد 00 و 10 و 11 وجود داشته باشد، از آن به عنوان یک کرانه­ی 01 نام برده می­شود. در غیر این صورت اگر بین دو کد فقط کلمه­های کد 00 و 01 و 11 وجود داشته باشد به آن یک کرانه­ی نوع 10 می­گویند. تمامی کرانه­­های موجود در یک کلمه کد غیرهمپوشان از نوع 10 یا 01 هستند. در کتابچه­های کد غیرهمپوشان با اندازه­ی بیشینه همواره دو شرط صادق است: 1) کتابچه­های کد غیرهمپوشان دارای کرانه­های یک در میان 10 و 01 است. 2) الگوهای 010، 101، 1001 و 0110 نمی­توانند در کتابچه­های کدگذاری غیرهمپوشان ظاهر شوند. حداکثر ظرفیت یک کتابچه­ی کد غیرهمپوشان با عرض m بیت از رابطه­ی 5 بدست می­آید:

در رابطه­ی 5 مقادیر پنج جمله­ی اول به ترتیب 2، 3، 4، 5 و 7 است [3].

# کارهای قبلی انجام شده

برای مقابله با همشنوایی از انواع کدهای اجتناب از همشنوایی در کدگذار-کدگشاهای تعبیه­شده در شبکه بر روی تراشه برای ارسال داده درون تراشه استفاده می­شود. زیرا که با پیشرفت فناوری معماری­های شبکه بر روی تراشه، سربار تحمیلی این­گونه کدگذار-کدگشاها از نظر مساحت، تأخیر وتوان بسیار مهم ارزیابی شده است.

در این مقاله از نرم­افزار Hspice در فناوری 130 نانومتر شرکت TSMC استفاده شده است. پارامترهای مورد نظر این مقاله مساحت و توان مصرفی بوده است. با توجه به شبیه­سازی­های انجام شده، مساحت ذخیره­شده با پهنای­باند متفاوت در دو کدِ ارائه شده به ترتیب زیر بوده است:

# نتیجه­گیری

امروزه برای مقابله با همشنوایی از انواع کدهای اجتناب از همشنوایی در کدگذار-کدگشاهای تعبیه­شده در شبکه بر روی تراشه برای ارسال داده درون تراشه استفاده می­شود.

# فهرست مراجع

|  |  |
| --- | --- |
| [1] | B. Fu and P. Ampadu, Error Control for Network-on-Chip Links, Springer Science & Business Media, 2011, p. 162. |
| [2] | م. مدرسی, “افزایش کارآیی و کاهش توان مصرفی شبکه های روی تراشه با استفاده از بازپیکربندی,” دانشگاه صنعتی شریف, تهران, 1389. |
| [3] | ا. پاطوقی, “كاهش اثر خطاي نرم و اشكال همشنوايي در شبكه هاي روي تراشه,” دانشگاه صنعتی شریف, تهران, 1389. |
| [4] | M. Mutyam, "Fibonacci Codes for Crosstalk Avoidance," IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS, 2012. |
| [5] | F. Shi, X. Wu and Z. Yan, "New Crosstalk Avoidance Codes Based on a Novel Pattern Classification," *IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS,* 2012. |
| [6] | K. Sinha, R. Ghosh and B. P. Sinha, "A New Number System Using Alternate Fibonacci Numbers as the Positional Weights with Some Engineering Applications," Springer, 2014. |
| [7] | X. Wu and . Z. Yan, "CAC CODEC DESIGNS BASED ON NUMERAL SYSTEMS," in *IEEE Workshop on Signal Processing Systems*, 2009. |
| [8] | X. Wu and Z. Yan, "Efficient CODEC Designs for Crosstalk Avoidance Codes Based on Numeral Systems," *IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS,* April,2011. |
| [9] | S. R. Sridhara, A. Ahmed and N. R. Shanbhag, "Area and Energy-Efficient Crosstalk Avoidance Codes for On-Chip Buses," in *Computer Design: VLSI in Computers and Processors*, 2004. |
| [10] | W.-W. Hsieh, P.-Y. Chen and C.-Y. Wang, "A Bus-Encoding Scheme for Crosstalk Elimination in," *IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS,* vol. 26, p. 12, 2007. |
| [11] | K. Ramesh and E. Srinivas, "FPGA Implementation of Codec Design for Optimal," in *International Journal of Science and Research (IJSR)*, February 2013. |
| [12] | C. Duan, . K. Gulatit and S. P. Khatrit, "mory-based Cross-talk Canceling CODECs for On-chip Buses," in *Circuits and Systems, 2006. ISCAS 2006. Proceedings. 2006 IEEE International Symposium on*, 2006. |
| [13] | C. Duan, C. Zhu and S. P. Khatri, "Forbidden Transition Free Crosstalk," in *ACM/IEEE Design Automation Conference*, 2008. |

1. 1 Very Large Scale Integeration [↑](#footnote-ref-1)
2. 2 System [↑](#footnote-ref-2)
3. 3 Chip [↑](#footnote-ref-3)
4. 4 System On Chip [↑](#footnote-ref-4)
5. 1 Switch [↑](#footnote-ref-5)
6. 2 Router [↑](#footnote-ref-6)
7. 3 Links [↑](#footnote-ref-7)
8. 1 Filit [↑](#footnote-ref-8)
9. 2 Starvation [↑](#footnote-ref-9)
10. 3 Wormhole [↑](#footnote-ref-10)
11. 1 Shielding [↑](#footnote-ref-11)
12. 2 Active Shielding [↑](#footnote-ref-12)
13. 3 Passive Shielding [↑](#footnote-ref-13)