علوم و مهندسی کامپیوتر
نشریه علمی پژوهشی انجمن کامپیوتر ایران
مجلد ۱، شماره ۱ (الف)، بهار ۱۳۸۲
صفحات ۳۸- ۴۶

نشریه علمی پژوهشی انجمن کامپیوتر ایران

مروری بر موفقیت ها و موانع در طراحی مدارهای ناهمگام

سیدولی ا... فاطمی و حسین پدرام

دانشکده مهندسی کامپیوتر، دانشگاه صنعتی امیر کبیر

چکیده

طراحی مدارها به روش ناهمگام از سالهای ۱۹۵۰ میلادی مطرح بوده است، ولی بدلیل پیچیدگی زیاد طراحی این مدارها و محدودیتهای تکنولوژی و ابزارهای خودکار طراحی، پیشرفت زیادی در این زمینه تا سالهای اخیر وجود نداشته است. اما با توجه به رشد تکنولوژی وطراحی و پیاده سازی ابزارهای طراحی بهینه این مدارها در سطح دنیا، انتظار میرود در آینده معماری ناهمگام جایگزین خوبی برای معماری همگام در برخی از کاربردها باشد. این پیشرفتها در سالهای اخیر منجر به ساخت اولین سری پردازنده‌های ناهمگام بصورت تجاری شده است. در این مقاله مروری بر انواع روشها، اقدامات انجام شده و موانع و پیشرفتهای بدست آمده در طراحی مدارها با این روش ارائه شده است.

کلمات کلیدی: مدارهای ناهمگام، طراحی خودکار، معماری ناهمگام

۱- مقدمه

در طراحی مدارهای منطقی ۲ فرض تاکنون مورد نظر بوده است : ۱) همه سیگنالها باینری هستند. ۲) زمان اجرای مدار منقطع می‌باشد. بکمک فرض اول امکان استفاده از منطق دو مقداره فراهم شده و با فرض دوم مشکل تغییرهای ناخواسته و رفتارهای پویای مدارهای دارای بازخور قابل صرفنظر کردن می باشد. مشخص است که حذف فرضهای فوق امکان ارائه نتایج بهتری را می‌توانند بوجود آورد. در مدارهای ناهمگام فرض انقطاع زمانی حذف شده ولی همچنان سیگنالها باینری فرض می‌شوند. این امر میتواند مزایای زیر را شامل شود:

حذف Clock Skew: اختلاف زمانی در رسیدن پالس ساعت به قسمت‌های مختلف مدار در مدارهای همگام محدودیتهایی در طراحی و سرعت سیستم بوجود می‌آورد که در مدارهای ناهمگام بعلت عدم وجود پالس ساعت سراسری نگرانی از این مشکل وجود نخواهد داشت.

توان مصرفی کمتر: در مدارهای استاندارد همگام در هر پالس ساعت یک تغییر (در واقع یک شارژ و یک تخلیه بار خازنهای تمامی گیتهای مدار حتی قسمتهایی که در تعیین خروجی نقشی ندارند) وجود خواهد داشت. بطور مثال مدارهای محاسبه ممیز شناور را می‌توان اشاره کرد که حتی در صورت عدم اجرای دستور ممیز شناور درپردازنده در هر پالس ساعت یک تغییر خواهند داشت. اگر چه ممکن است مدارهای ناهمگام در قسمتهای در حال محاسبه نیاز به تغییر بیشتری داشته باشند ولی بعلت وجود همه تغییرها فقط در محدوده درگیر در محاسبات، توان کمتری در مجموع مصرف خواهد داشت.

ارائه کارایی متوسط بجای بدترین کارایی مدار: در مدارهای همگام لازم است پس از خاتمه محاسبات منتظرتکمیل همه تغییرها شده سپس مرحله بعدی عملیات را شروع نمود که این حالت را بدترین کارایی قابل ارائه مدار تعریف می‌کنند. در صورتیکه در مدارهای ناهمگام پس از پایان هر مرحله عملیات مرحله بعد بلافاصله می‌تواند شروع شود.

عدم نیاز به بهینه‌سازی سراسری مدار: در مدارهای همگام بعلت وابستگی پالس ساعت مدار به کندترین مسیر محاسباتی لازم است بهینه سازهایی برای حصول پالس ساعت سریعتر و کارایی بهتر برای هر مسیر محاسباتی اعمال کرد. در مدارهای ناهمگام سرعت سیستم فقط به مسیر مدار در حال عملیات وابسته بوده و نیازی به بهینه‌سازی سراسری مدار نخواهد بود.

دارای پتانسیل بهتر برای استفاده از تکنولوژیهای پیاده‌سازی جدید: عموماً مدارهای مجتمع در طول حیات خود با چندین تکنولوژی پیاده‌سازی می‌شوند. بدلیل عدم وابستگی تأخیر قسمتهای مختلف مدار ناهمگام، حتی می‌توان فقط قسمتی از مدار را نیز با تکنولوژی سریعتر پیاده‌سازی و تعویض نمود.

تطبیق اتوماتیک با خصوصیات فیزیکی محیط و تولید: تأخیر یک مدار با تغییر پارامترهای تولید و ولتاژ منبع تغذیه تغییر خواهد کرد. برای یک مدار همگام برای محاسبه سرعت سیستم لازم است بدترین حالت برای این پارامترها در نظر گرفته

شود. در مدارهای ناهمگام با توجه به تولید سیگنال پایان عملیات در همان شرایط محیطی می‌توان بلافاصله بعد از پایان واقعی کار قبلی به عملیات ادامه داد.
دارای پایداری بیشتر برای مدارهای دسترسی انحصاری و کنترل ورودی‌های خارجی: عناصر تضمین کننده دسترسی انحصاری سیگنالهای مستقل و سیگنالهای خارجی با پالس سراسری مدار از منابع مهم ایجاد پایداری ضعیف در مدارها می‌باشند. [۳۱] یک وضعیت نیمه پایدار شامل وضعیت ناپایداری مثل اتصال ضربدری دو عنصرNOT در ولتاژ ۲/۵ ولت می‌باشد. در این وضعیت مدار برای مدت نامحدود خواهد ماند [۳۲]. بنابراین طراحی این مدارها در معماری همگام باعث بروز اشکال در مدار خواهد شد. در صورتیکه در مدل ناهمگام مدار می‌تواند در یک وضعیت برای تکمیل شدن تا هر مدتی منتظر مانده و دسترسی انحصاری پایدار را نتیجه دهد. همچنین با توجه به عدم نیاز به هماهنگی با پالس سراسری، در مدارهای ناهمگام ورودی‌ها بسیار ساده‌تر از دنیای خارج قابل دریافت هستند.

با توجه به تمامی مزایای فوق این سؤال مطرح خواهد بود چرا سیستمهای همگام کنار گذاشته نمی‌شوند. دلیل این امر وجود مشکلات فراوان در طراحی مدارهای ناهمگام می‌باشد. بطورکلی طراحی مدارهای ناهمگام بصورت یک روش سریع و ساخت‌یافته بسیار دشوارتر از مدارهای همگام می‌باشد. در سیستم همگام، طراح با تعریف سریع مدار ترکیبی مورد نیاز و قراردادن تعدادی ثبات در اطراف آن و لحاظ پالس ساعت منطقی، هیچ نگرانی از تغییرهای ناخواسته سیگنالها یا حالتهای پویا و ناخواسته مدار نخواهد داشت. برعکس طراح یک مدار ناهمگام برای جلوگیری از ارائه نتایج غلط لازم است دقت بسیار زیادی در حالتهای پویایی مدار و حذف یا جلوگیری از آنها در زمان طراحی داشته باشد. همچنین رعایت ترتیب عملیات که بوسیله یک ثبات در مدار همگام لحاظ می‌شود لازم است کنترلی در مدارهای ناهمگام لحاظ گردد. برای سیستمهای پیچیده رعایت ترتیب دستور العملها بصورت دستی غیرممکن خواهد بود. متأسفانه بصورت عمومی امکان طراحی مدارهای ناهمگام با استفاده از ابزارهای طراحی مدارهای همگام یا شبیه مدارهای همگام و با پیاده‌سازی متفاوت موجود نمی‌باشد. بعنوان مثال بعضی روشهای طراحی ناهمگام یکسری توابع جبری (اشتراک‌پذیری، جابجایی و قانون دمورگان) را برای ساده‌سازی مدار مجاز نمی‌دانند. همچنین ابزارهای CAD مکان‌یابی و راهیابی و تقسیم‌بندی موجود مدارهای همگام نیز برای استفاده این مدارها نیاز به تصحیح دارند.
وجود تمامی مشکلهای فوق و نیاز به رفع آنها، طراحی ناهمگام را به عنوان موضوعی بسیار مهم در محیط‌های تحقیقاتی مطرح می‌سازد. این امر مستقل از موفقیتهای سیستمهای همگام می‌باشد. مدارهای ناهمگام براحتی برای ارتباط محیطهای همگام با محیط و یا بین سیستمهای همگام مورد استفاده قرار میگیرند. مشکلات اشاره شده باعث شده است علیرغم مطرح بودن این تکنولوژی از سال ۱۹۵۰ میلادی تاکنون استفاده فراوان و رشد قابل ملاحظه‌ای در بکارگیری آنها در محیطهای واقعی مشاهده نشود. علیرغم تمامی موارد فوق بیشترین مزیت مدارهای ناهمگام مقاوم بودن در برابر تغییرات محیطی است و ارائه سرعت بیشتر در عمل قطعی نمی‌باشد، زیرا که در مدارهای ناهمگام با توجه به سیاست هماهنگ‌سازی سیگنالها، زمان بیشتری بصورت عمومی مورد نیاز خواهد بود. با توجه به تمامی این ابهامات و مستقل از موفقیتهای روشهای همگام، طراحی ناهمگام موضوع مهمی در تحقیقات جدید می‌باشد. همچنین ممکن است این مدارها بعنوان رابط با محیط در سیستمهای همگام نیز مورد استفاده قرار گیرند.
در این مقاله با هدف ارائه کلیات مشکلات موجود، روشهای طراحی این مدارها مورد بررسی قرارگرفته‌اند. این روشها بر مبنای مدل در نظر گرفته شده برای تأخیر عناصر و سیمهای مدار تقسیم‌بندی میشوند که در بخش ۲ به آنها اشاره شده است. با توجه به تفاوت ماهیت عملکرد و نحوه طراحی قسمتهای کنترلی و مسیر داده در مدارهای ناهمگام در بخش ۳ روشهای طراحی این مدارها بر این مبنا دسته‌بندی و ارائه شده اند. در بخش ۴ معروفترین روشها و ابزارهای استفاده شده در پردازنده‌های ناهمگام موجود مورد بررسی و مقایسه قرار گرفته و در بخش ۵ آخرین پردازنده‌های طراحی شده در این معماری ارائه و در حد امکان با مشابه همگام خود مورد مقایسه قرار گرفته اند.

۲- انواع مدلهای تاخیر در مدارهای ناهمگام

الف- مدارهای با تأخیر محدود (Bounded Delay)

در این مدارها تأخیر در تمام عناصر و سیمها، معلوم و یا حداقل محدود فرض می‌شود. بکمک جداول حالت طراحی این مدارها انجام می‌شود و لازم است نکات لازم جهت جلوگیری از حساسیت به تقدم سیگنالها در جدول توسط طراح در نظر گرفته شود. نکته اثبات شده در این روش عدم تولید تغییر ناخواسته جدید توسط قوانین توزیع‌پذیری و دمورگان جهت ساده‌کردن مدار می‌باشد. از محدودیتهای این روش مجاز بودن تغییر فقط در یک ورودی در هر لحظه برای کنترل تغییر ناخواسته می‌باشد. همچنین از راههای کنترل تغییر ناخواسته ترتیبی اضافه نمودن تأخیر مناسب در مسیرهای بازخور می‌باشد که در کارآیی تأثیر خواهد داشت.

ب- مدارهای مستقل از سرعت (Speed Independent, SI)

در این مدل تأخیر گیتها نامحدود و تأخیر سیمها بسیار کم و قابل صرفنظر در مقایسه با تأخیر گیت در نظر گرفته می‌شود. دراین روش مدار به دو قسمت کنترلی و داده تقسیم شده و برای طراحی و پیاده‌سازی هر قسمت، از تکنیکهای متفاوتی استفاده می‌شود. برای طراحی مسیر داده معمولاً از کدهای خودهمساز یا زوجی استفاده می‌شود. بدیهی است استفاده از هر کدام از روشهای فوق خطر بروز تغییر ناخواسته در مسیر داده را از بین برده ، اجازه می‌دهندخطوط داده با هر ترتیب زمانی تغییر یابند. در طراحی قسمت کنترل برای جلوگیری از تغییر ناخواسته از تغییر (تحریک) ورودی قبل از پایدار شدن کامل خروجی و تأثیر در قسمت بعدی مدار جلوگیری می‌شود. با پیشرفت تکنولوژی و افزایش سرعت گیتها از تأخیر سیمها در مقابل گیتها براحتی نمی‌توان صرفنظر نمود.

ج- مدارهای غیرحساس به تأخیر (Delay Insensitive, DI)

در این روش برخلاف روش تأخیر محدود، تأخیر در سیمها و عناصر یا المانهای مدار نامحدود در نظر گرفته می‌شود. در این مدل بدلیل فقدان هیچ ضمانتی برای زمان دریافت و ارسال اطلاعات، لازم است فرستنده و گیرنده بکمک سیگنالهایی ارسال و دریافت را به هم اطلاع دهند. این عمل بکمک مدار تشخیص سیگنال پایان عملیات در گیرنده انجام خواهد شد. در این حالت فرستنده نیز منتظر دریافت این سیگنال برای ارسال اطلاعات بعدی می‌باشد. در این روش مرحله طراحی و ساخت فیزیکی مدار هیچ وابستگی به هم نداشته و درستی عملکرد مدار وابستگی به تأخیر عناصر مدار نخواهد داشت.
با وجود جذاب بودن ایده فوق، این ایده دارای دو ایراد اساسی می‌باشد. اولین ایراد اینست که گیتهای معمولی مانند OR و XOR و AND که دارای چند ورودی و یک خروجی می‌باشند (به استثناء Buffer و NOT وC-element) دارای مشخصه عدم حساس به تأخیر نیستند. برای طراحی مدارها بر طبق مدل مذکور به یک سری گیتهای خاص با چند خروجی نیاز می‌باشد. علاوه بر این در ساخت چنین گیتهایی نیز مجبور به داشتن فرضهایی در مورد تأخیر داخلی آنها می‌باشیم. به عبارت دیگر می‌توان گیتها را چنان ساخت که مشخصه عدم حساس به تأخیر را دار ا باشند و با محیط به صورت عدم حساس به تأخیر ارتباط پیدا نمایند ولی ساخت خود گیتها بر اساس مدل تأخیر محدود صورت گیرد. البته این مشکل قابل رفع است و چنین گیتهایی تاکنون ارائه شده‌اند.
ایراد دوم عدم امکان طراحی و ساخت بسیاری از مدارها با این مدل می‌باشد. به عبارت دیگر این مدل کامل نمیباشد. به عنوان مثال همچنانکه خود Udding [۶]

اثبات نموده است، مدار یک داور منصف را نمی‌توان براساس مدل عدم حساس به تأخیر طراحی نمود. داور مداری می‌باشد که یک منبع مشترک را در هر زمان به یک متقاضی اختصاص می‌دهد و اگر همواره منبع را به یکی از متقاضیان اختصاص داده و تقاضاهای دیگری را نادیده بگیرد منصف نیست. مشکل فوق تنها از طریق تضعیف تعریف مدل قابل حل می‌باشد بطوریکه مدارهای حساس به ترتیب دریافت سیگنالها را بتوان با آن مدل طراحی نمود.

۳- دسته بندی مدارهای ناهمگام

الف - کنترل کننده های ناهمگام

مهمترین مسئله در طراحی کنترل کننده های ناهمگام جلوگیری از امکان بروز تغییر ناخواسته در مدار می باشد. بدلیل عدم وجود پالس سراسری برای بی‌اثرکردن تغییرات ناخواسته در مدار (شبیه مدارهای همگام) هر گونه تغییر ناخواسته‌ای می‌تواند بعنوان تغییر واقعی تلقی و عملکرد کنترل کننده و یا مدار را تحت تأثیر قرار دهد. عموماً روشهای متعارف کنترل تغییر ناخواسته یا در رفتار مدار تأثیرمیگذارند و یا کارائی مدار را تحت تأثیر قرار میدهند. بنابراین به روشهای اصولی در زمان طراحی مدار برای کنترل تغییر ناخواسته نیاز خواهد بود. دسته‌بندی روشهای طراحی مدار یا نحوه کنترل تغییر ناخواسته شامل تبدیل لغوی، مبتنی بر گراف و FSM میباشد:

روش تبدیل لغوی: عملیات سنتز در این روش شامل تبدیل کلمه‌ای یا ترجمه و یا تبدیل جبری توصیف سطح بالای مدار به مدار DI یا SI میباشد. از مزایای این روش امکان توصیف سیستمهای موازی پیچیده با زبانهای سطح بالا (بدون محدودیت و جزئیات توصیف سطوح پایین) و تبدیل مرحله‌ای آن را میتوان اشاره کرد. عدم بهینه‌سازی سراسری و کامل در طراحی خودکار و ارائه نتایج غیرمفید مثل مساحت زیاد مدار از جمله معایب این روش می باشد. از جمله این روشها می‌توان از روش Martin بر مبنای زبان CSP، روش Brunvand بر مبنای زبان OCCAM، روش Ebergen بر مبنای Trace Theory و زبان TANGRAM نام برد.

روش مبتنی بر گراف: تقریباً اغلب این روشها از Petri Net یا نوعی از آن برای بیان مشخصات مدار استفاده می‌کنند. قابلیت بیان همزمانی عملیات از جمله مزایای این روش و پیچیدگی توصیف انتخاب ورودیها یا انتخاب رفتار بعدی بر اساس تغییرات ورودی مدار از جمله معایب این روش میباشد. اغلب روشهای این دسته، مدار را در مدل SI تولید می‌کنند. برای رفع یا کنترل تغییر ناخواسته عموماً یک تأخیر داخلی بر مبنای مدل تاخیر محدود به عنصر اضافه میشود که کارائی مدار را کاهش میدهد.

روش مبتنی بر FSM: در روشهای اولیه پس از یک تغییر لازم بود تا پایداری مجدد مدار برای اعمال ورودی بعدی منتظر بود در حالیکه در روشهای اخیر امکان اعمال چندین تغییر همزمان ورودی موجود میباشد. این روشها که Burst Mode Machine یا بعبارتی ماشین Mealy را پیاده‌سازی می‌کنند اجازه چندین تغییر را در یک وضعیت میدهند. به این معنی که با تغییر همه ورودیها، ماشین مجموعه خروجی‌ها را تولید و سپس به وضعیت جدید میرود. انتخاب ورودی در این روش ساده‌تر از روش قبلی بوده و همچنین کددهی حالات مدار نیز ساده‌تر میباشد. این روش در توصیف و پیاده‌سازی مدارهای بزرگ مثل کنترل‌کننده حافظه نهان و یا کنترل‌کننده ارتباطات مفید میباشد. از جمله مدارهای طراحی شده با این روش Differential Equation Solver توسط Ted Williams و یا High Performance SCSI Controller و یا Intel Asynchronous Instruction Length Decode Chip را میتوان نام برد.

ب - مسیرهای داده ناهمگام

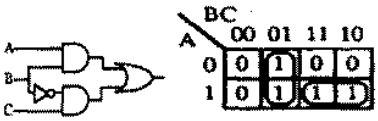
از جمله عناصر مهم در طراحی ناهمگام جمع‌کننده‌ها، ضرب‌کننده‌ها، **حافظه‌ها**، و عوامل خط لوله میباشد. با توجه به ویژگی شاخص عناصر ناهمگام در ارائه سیگنال پایان عملیات نسبت به عناصر همگام و امکان اجرای متوالی عملیات بدون نیاز به زمان مرده بین عملیات، طراحی این عناصر از اهمیت بیشتری برخوردار میباشد. معروفترین روش تشخیص پایان عملیات استفاده از سیگنالهای زوجی میباشد. روش دیگر اندازه گیری جریان یا مانیتورکردن جریان مصرفی منبع تغذیه میباشد که دارای پیچیدگی بیشتری میباشد. در حال حاضر انواع جمع‌کننده‌ها و یا ضرب‌کننده‌های ناهمگام با کارائی خوب طراحی و ساخته شده اند. از جمله میتوان به ساخت جمع‌کننده ۳۲ بیتی با تکنولوژی ۰/۶ میکرومتر CMOS و با تأخیر متوسط کمتر از ۲ نانو ثانیه یا ضرب‌کننده ۳۲ بیتی با تأخیر ۸ نانو ثانیه و یا تقسیم کننده ۵۴ بیتی Williams با ۱۶۰ نانو ثانیه اشاره کرد. بهترین طراحی حافظه نهان در حال حاضر در پردازنده Amulet 2e ارائه شده است.

استفاده از خط لوله‌ای ریز با عمق بیش از ۲۴ در پردازنده ناهمگام MIPS R3000 توسط گروه MARTIN در دانشگاه Caltech و با ارائه کارایی فوق‌العاده در محاسبات جمع و ضرب در مسیر داده، نمایش خوبی از کارایی این تکنیک در معماری ناهمگام می‌باشد. در حال حاضر تحقیقات زیادی برای همپوشانی تأخیرهای مدار و استفاده مؤثرتر از المانها بکمک این تکنیک در دست انجام میباشد.

۴- مروری بر روشهای معروف طراحی مدارهای کنترل‌کننده ناهمگام

الف - Foundamental Mode (FM) Huffman Circuits

طراحی مدار در این روش خیلی شبیه مدارهای همگام می‌باشد [۳۳]. عموماً از جدول حالات برای بیان حالتهای مدار استفاده می‌شود. در این جدول ردیفها متناظر وضعیتهای داخلی و ستونها ترکیبهای ورودی را شامل می‌شوند. ورودی هر خانه وضعیت جدید و خروجی را نمایش می‌دهد. حالتهایی که وضعیت جدید با خودشان متناظر باشد حالتهای پایدار در مدار خواهند بود. از نکات قابل توجه در پیاده‌سازی سیستم‌های غیر همگام به این روش، حالتهای میانی ناشی از چندین تغییر ورودی می‌باشد. بعنوان مثال برای تغییر ورودی از ۰۰ به ۱۱ در جدول حالت شکل ۱ تغییر ورودی به ۰۱ یا ۱۰ اجتناب ناپذیر است. بنابراین لازم است در وضعیت ۱ بر ای ورودیهای ۰۱ و ۱۰ نیز سیستم در وضعیت ۱ باقی بماند.

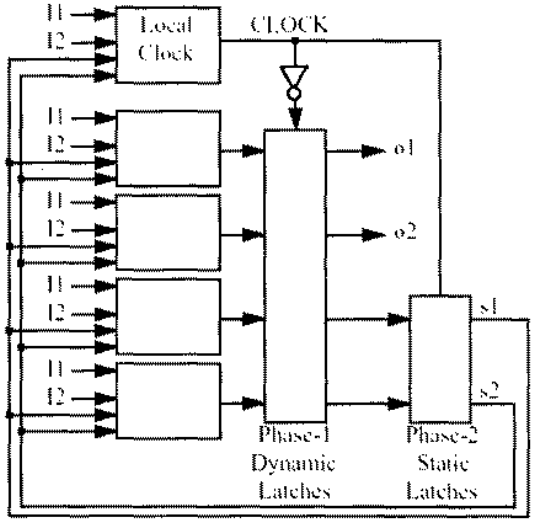


شکل ۱- هازارد در FM

حذف رفتارهای ناخواسته مثل Hazard در این روش بر عهده طراح خواهد بود. بعنوان مثال در جدول فوق لازم است تعدادی ترکیب اضافی برای جلوگیری از Static Hazard (مثل AC) به مدار اضافه گردد. همچنین بعنوان مثال برای تغییر از حالت (۰و۱و۱) به (۱و۰و۱) در صورتی که ورودی C دچار تأخیر شود وضعیت میانی (۰و۰و۱) و خروجی ناخواسته ۰ خواهیم داشت. برای این حالت لازم است طراح با اضافه نمودن تأخیر مدار را به حالت میانی (۱و۱و۱) ببرد تا در خروجی هیچ تغییر ناخواسته‌ای بروز ننماید. از مشکلات عمده این روش افزایش تعداد ورودیهای گیتهای AND یا OR با بزرگ‌شدن مدار جهت کنترل تغییرهای ناخواسته می‌باشد که نیاز به شکستن آنها خواهد بود. از جمله قوانین قابل استفاده تمامی توابع جبری مدل همگام مثل اجتماع، اشتراک و دمورگان بوده که باعث ایجاد تغییرهای ناخواسته جدید در مدار نمی‌شوند. اطمینان از پایداری مدار بازای

هر تغییر ورودی قبل از تغییر بعدی از نکات لازم در این طراحی می‌باشد. این امر بکمک افزایش تأخیر در مسیر بازخور مدار و محدودیت تغییر یک بیت وضعیت جدید قابل حصول می باشد. استفاده از کدگذاری one-hot از جمله راههای پیاده‌سازی برای اعمال محدودیت فوق می باشد.

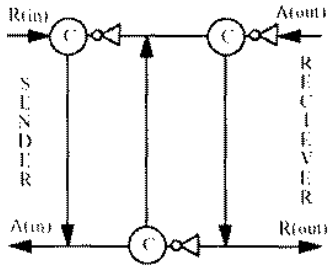
راههای متفاوتی برای کنترل ورودی مدار ارائه و پیاده‌سازی شده است که همگی علاوه بر پیچیدگی طراحی در صورت افزایش مقدار ورودی پیاده‌سازی آنها غیر مقدور خواهد بود. از جمله راههایی که برای اجازه چندین تغییر همزمان ورودی توسط Nowick [۳۴] ارائه شده اضافه نمودن یک ثبات میانی برای ثبت لحظه‌ای تغییرات و اعمال همزمان آنها به مدار می‌باشد که حالتهای میانی ناشی از تأخیر را می تواند از بین ببرد. نیاز به پالس ساعت محلی و ثبات اضافی جزئی از مشکلات عمده این روش می باشد (شکل ۲)



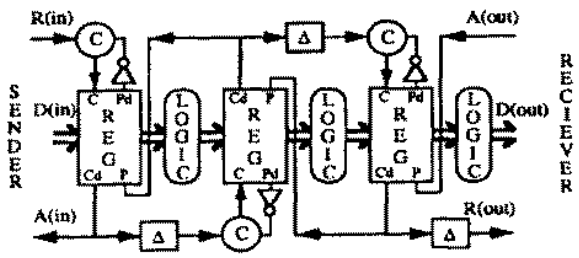
شکل ۲ – مدار آسنکرون با پالس ساعت محلی [۳۴]

در مجموع پیچیده بودن اغلب ماشینهای حالت، وجود مسیرهای داده و پیچیدگی تحمل خرابی مشکل عمده طراحی با این روش می باشد که کاربرد این روش را محدود کرده است.

ب- Micropipeline



شکل ۳ (الف)- کنترل مایکروپایپ لاین [۳۵]



شکل ۳ (ب)- جریان داده مایکروپایپ [۳۵]

در ساختار خط لوله شکل ۳ ترکیبی از ویژگیهای مدل تأخیر محدود و غیر حساس به تأخیر استفاده شده است [۳۵]. در این ساختار (شبیه بافر)، درخواست یک یا تغییر از طریق R_{in} وارد و از طریق R_{out} خارج می‌شود. با درخواست R_{in} و در صورت آماده بودن واحد بعدی (سیگنال Ack از واحد بعدی) امکان ورود داده جدید به یک طبقه فراهم خواهد بود. این آمادگی پس از زمان تأخیرD، متناظر با زمان محاسبه در مدار ترکیبی، به واحد بعدی اطلاع داده می‌شود. عنصر با برچسب C به المان مولر معروف است و در صورتیکه تمامی ورودیهای آن صفر یا یک باشد خروجی یک و در غیر این صورت خروجی صفر تولید می‌کند. یک سیگنال ورودی از Rin پس از عبور از تمامی عناصر C بصورت سری از طریق Rout از خط لوله خارج می‌شود.در طول این مسیرچرخه هر مرحله بصورت مستقل شامل عناصرC و منفی کنندهٔ طی می‌شود. این سیگنال در صورت ورود سیگنال Aout امکان ورود به طبقه آخر را خواهد داشت. بنابراین در صورت آماده نبودن خروجی سیگنال ورودی دیگری نیز به خط لوله وارد نخواهد شد. این امر بدلیل نیاز فرستنده به سیگنال آمادگی Ain دارد که در صورت عدم آمادگی طبقه خروجی (تولید نشدن C و Cd) ثبات طبقه خروجی برای فرستنده نیز ارسال نمی‌گردد. نقش عمده عنصر C هماهنگی آماده شدن ورودی و خالی بودن هر طبقه خط لوله برای ورود اطلاعات جدید می باشد.

مشاهده می‌شود که در این مدل در قسمت پرداز ش داده تغییر ناخواسته‌های مختلف می‌تواند رخ دهد ولی پروتکل تأخیر محدود آنها را بی أثر می‌نماید. یعنی داده‌ایکه در هر رجیستر ذخیره می‌گردد با توجه به تأخیر مسیرسیگنال Req مطمئناً درست و عاری از هرگونه تغییر ناخواسته خواهد بود. می‌بینیم که مشکل اصلی مدارهای غیرهمگام که همان بروز تغییر ناخواسته‌های مختلف بود در این مدل بسادگی حل شده است. علاوه بر این طراحی بر مبنای آن بسیار ساده می‌باشد و پیچیدگی مدلهای قبلی موجود نخواهد بود. همین مسأله باعث شده است که این مدل در سالهای پیش بسیار مورد توجه قرار گیرد بطوریکه تاکنون طراحی مدارهای مختلفی از جمله چند پروسسور بر مبنای آن انجام گرفته است.

همانطوریکه مشاهده شد اگر چه این طراحی پیاده‌سازی موفقی برای خط لوله پویا بوده و مشکل تغییرهای ناخواسته خروجی ثباتها را حل نموده است ولی کارائی ارائه شده بدترین حالت کارائی می‌باشد که بدلیل افزایش تأخیرهای اضافی در مسیر کنترل مدار برای تطبیق بدترین زمان محاسباتی مدار ترکیبی می‌باشد. خط لوله اشاره شده فقط مسیر مستقیم داده را شامل شده و نیاز کنترل خط لوله های موازی یا دارای مسیرهای بازخور در پردازش سیگنال‌ها در ماشینهای حالت دچار مشکل می‌باشد.

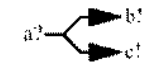
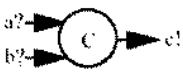
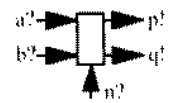
از جمله پردازنده‌های طراحی شده با این روش را می‌توان Amulet در دانشگاه Manchester نام برد. این پردازنده علیرغم حفظ تمامی خواص مدارهای غیرهمگام کارایی خوبی را تاکنون ارائه نداده است.

ج- Trace Theory و روش مارتین

TraceTheory [۲۶] یک تئوری ریاضی می‌باشد که در آن از مدل یکسانی برای توصیف بلوکها و المانهای مدار استفاده می‌کند. این مدل می‌تواند برای بیان عملکرد سیستمهای دینامیک و همچنین توصیف عملکرد مدارهای غیرهمگام می‌تواند بکار گرفته شود. مزیت استفاده از این ابزار زمینه تئوریک بسیار قوی آن می‌باشد به طوریکه این مسئله باعث مورد توجه قرار گرفتن آن بطور بسیار وسیع گشته است.

ساختار یک Trace به صورت جمله معمولی است که بصورت <tT و aT> نمایش داده می‌شود که در آن aT یک مجموعه متناهی از سمبولها معادل سیگنال مدار می‌باشد و به آن alphabet نیز گفته می‌شود، tT عبارت است از یک مجموعه که اعضای آن خود یک سری متناهی از سمبولهای متعلق به aT می‌باشند. در این روش برخلاف روشهای قبلی، از مدل یکسانی شبیه عبارات منطقی برای بیان

Name	Syntax	Meaning	Example
Input	<sym> ?	<sym> ∈ Input Alph & occurs in string	a? = {"a"}
Output	<sym> !	<sym> ∈ Output Alph & occurs in string	a! = {"a"}
Concatenation	<cmd1> ; <cmd2>	<cmd2> follows <cmd1>	a; b = {"ab"}
Union	<cmd1> | <cmd2>	either <cmd1> or <cmd2>	a | b = {"a", "b"}
Repetition	* [<cmd>]	zero or more concatenations of <cmd>	*[a] = {ε, "a", "aa", ...}
Prefix-closure	pref <cmd>	Any prefix of <cmd>	pref("ab") = {ε, "a", "ab"}
Projection	<cmd>↓<alph>	Remove all symbols from <cmd> not contained in <alph>	abc↓{a, c} = {"ac"}
Weave	<cmd1> ‖ <cmd2>	Shuffling of <cmd1> and <cmd2>, with shared symbols occurring simultaneously	abd ‖ acd = {"abcd", "acbd"}

Name	Specification	Schematic
Wire	pref *[a?; b!]	
IWire	pref *[b!; a?]	
Fork	pref *[a?; (b! ‖ c!)]	
C-element	pref *[(a? ‖ b?); c!]	
XOR	pref *[(a? | b?); c!]	
Toggle	pref *[a?; b!; a?; c!]	
Sequencer	pref *[a?; p!] ‖ pref * [b?; q!] ‖ pref * [n?; (p! | q!)]	
NCEL	pref *[$(a?)^2$ | $(b?)^2$ | ((a? ‖ b?); c!)^2]	
RCEL	pref *[$(a?; d!)^2$ | $(b?; e!)^2$ | ((a?; (d! ‖ c!)) ^{2} ‖ $(b?; (e! ‖ c!))^2$)]	

شکل ۵- مؤلفه های پایه ای Trace Theory [۲۴]

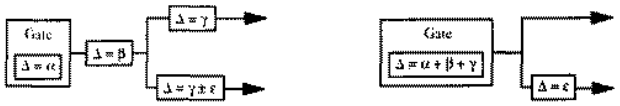
مشخصات بلوکها و طراحی مدار استفاده می‌شود. یک Trace شامل یکسری حروف الفبا و یکسری رشته که قابلیتهای مدار را توصیف می‌کند، می‌باشد. هر نماد حروف الفبا، متناظر با یک سیگنال در مدار و وجود یا ظهور یک نماد در یک جمله نمایش یک تغییر روی سیگنال می‌باشد. معمولاً حروف الفبا به دسته‌های مشخص برای بیان سیگنالهای ورودی، خروجی، داخلی و محیطی تقسیم شده است. همچنین از یکسری دستور برای سادگی نمایش قابلیتها شامل ورودی، خروجی، اتصال، اجتماع، تکرار و ترکیب استفاده می‌شود. در حالت عادی یک مدار فقط با ورودی و خروجی‌هایش توصیف می‌شود و در حین پردازش کامل مدار جزئیات داخلی و محیطی مورد استفاده خواهد بود. در این روش جهت سهولت بیان قابلیتهای مدار از یکسری فرمان ها در قالب یک جمله استفاده می شود. مجموعه این فرمانها در شکل ۴ اشاره شده است. از فرمانهای اشاره شده فقط فرمان همزمان‌سازی استفاده می‌شود فرمانهای جدیدی محسوب می‌شوند.

Projection که برای توصیف سلسله مراتبی و Weave که برای عملیات موازی و عناصر پایه می باشد تعدادی از این عناصر پایه در شکل ۵ نمایش داده شده است. همانطوریکه اشاره شد از خصوصیات جالب توجه این مدل توصیف یکسان مدار و با توجه به حضور عناصر با بیش از یک خروجی این محدودیت در این روش طراحی موجود نخواهد بود.

برای طراحی مدار بکمک این روش ابتدا رفتار ورودی، خروجی مدار بکمک دستورات Trace توصیف می‌شود. جهت کنترل تغییر ناخواسته یا تضمین عدم حساسیت دستورات به تأخیر یکسری قوانین و دستورالعملهایی زبانی تهیه شده است. روش فوق قادر به طراحی و ساخت مدارهای کاملاً غیر حساس به تأخیر می‌باشد ولی دارای مشکلات خاص خود است. اولاً استفاده از Trace Theory برای افراد عادی بسیار مشکل می‌باشد و این به دلیل پیچیدگی نسبی تئوری می‌باشد. این مشکل را می‌توان توسط زبانهای سطح بالا حل نمود. مشکل دوم حجم بسیار زیاد مدارهای طراحی شده در این روش می‌باشد به طوریکه ممکن است به غیرممکن بودن پیاده‌سازی بیانجامد.

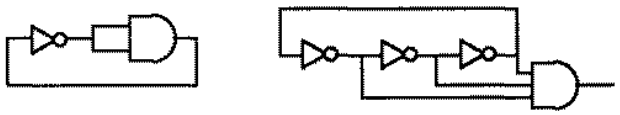
برای حل مشکل فوق Martin ایده لحاظ انشعابهای با تأخیر مساوی را بنام Isochronic Forks ارائه نموده است به این صورت که اگر بتوان در بعضی نقاط مدار، تأخیر دو شاخه منشعب از یک سیم را مساوی در نظر گرفت حجم مدار را به طور قابل ملاحظه‌ای می‌توان کاهش داد. انشعاب همگن به انشعابی گفته میشود که اختلاف تاثیر در مسیرهای انشعاب قابل صرفنظر کردن باشد. این محدودیت در مدارها غیر حساس به تاخیر مدل شبیه مدارهای مستقل از سرعت را در بهترین حالت نتیجه میدهد. در شکل ۶ مدار معادل مستقل از سرعت یک انشعاب دوتایی نمایش داده شده است. بکار گیری این ایده باعث انحراف از تعریف عدم‌حساس به تأخیر شده و درستی عملکرد مدار را نیز به نحوه پیاده‌سازی فیزیکی وابسته نموده، تست دقیق تأخیرها را ضروری می‌سازد. بدین ترتیب هدف اصلی از ارائه مدل عدم حساس به تأخیر کمتر برآورده می‌شود. مدارهای طراحی شده با این ایده را Quasi Delay Insensitive یا QDI می نامند. [۱، ۲، ۳، ۴، ۵ و ۶]

مدارهای شکل ۷ QDI بوده ولی DI نمیباشند. مدار سمت چپ در صورت وقوع تغییر در بالا رونده خروجی منفی کننده عنصر AND منتظر تغییر هر دو ورودی قبل از تغییر خروجی می‌ماند. در صورتی که برای حالت سیگنال پائین رونده عنصر AND بمحض دریافت اولویت بعد خروجی را تغییر می‌دهد. این مدار با توجه به



(الف)- isochronic fork (ب)- مدار معادل SI

شکل ۶- مدارهای معادل مستقل از سرعت



شکل ۷- مثال هایی از مدارهای QDI که DI نیستند.

ضمانت تغییر بقیه ورودی‌ها در هر شرایطی، QDI می‌باشد. عنصر OR نیز رفتار مشابه عنصر AND را دارد. در صورتیکه عناصر XOR یا XNOR QDI نمی‌باشند زیرا خروجی متناسب با اینکه یکی یا هر دو ورودی تغییر کند متفاوت خواهد بود. علیرغم اینکه مدلهای QDI و SI راه حلهای بیشتر و بهتری از DI را برای پیاده‌سازی ارائه می‌دهند ولی با توجه به الزامهایی که برای تأخیر مدار مطرح می‌کنند امکان پیاده‌سازی در همه محیطها را کم می‌کنند. بعنوان مثال در FPGA تأخیر سیستمها عموماً تأخیر گیتها را تحت‌الشعاع قرار می‌دهد. همچنین که تأخیرهای مرتبط با محیط اطراف عنصر شدیداً قابل توجه می‌باشد. اگرچه پیاده‌سازی QDI از SI راحتتر می باشد ولی همچنان پیاده سازی QDI نیز سخت می باشد. اختلاف طول سیمها و ساخت گیتها و سطح آستانه سوئیچها همگی از عواملی هستند که می توانند شرایط QDI را نقض نمایند.

در روش Martin که بسیار شبیه روش Ebergen [۸] می‌باشد، مدار توسط یک زبان سطح بالا شبیه CSP (Communicating Sequential Processes) توصیف می‌شود، این توصیف توسط کامپایلر به یکسری جمله ساده قابل پیاده‌سازی سطح سوئیچ تبدیل می‌شود. همچنین نقاطی که انشعاب در آنها باید isochronic باشد، مشخص می‌گردند. در این روش با اعمال یکسری عملیات تجزیه، پیچیدگی مدار کاسته و با بسط مدار توسط پروتکل چهار مرحله‌ای توصیف مدار به مجموعه‌ای از تغییرات تبدیل و با اعمال عملیات بر زدن و اضافه نمودن متغیرهای حالت از تولید ابهام در مدار جلوگیری می‌شود. ابهام در مواردی است که توصیف مدار می‌تواند

مراحل طراحی مدار به روش مارتین بطور خلاصه به شرح زیر می باشد:

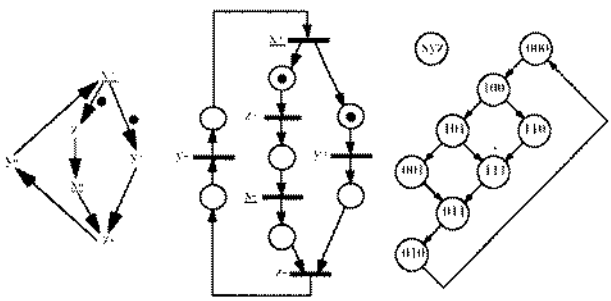
- شامل یک یا چند پروسس که بصورت موازی عمل کرده و از طریق کانال با هم ارتباط برقرار می‌کنند	Concurrent Program (Like CSP)
- شکستن به پردازه‌های کوچکتر	Process Decomposition
- قسمت داده می‌تواند بصورت استاندارد پیاده‌سازی شود ولی قسمت کنترل باید مرحله به مرحله طراحی گردد.	Separation of Control & Data Parth
- اعمال پروتکل ارتباطی (۴ مرحله‌ایی یا ۲ مرحله ایی) به ارتباط در کانال	Handshaking Expansion
- موکول کردن نیمه دوم پروتکل ارتباطی به زمان بعد	Reshuffling
- اضافه کردن متغیر میانی در صورت وجود ۲ وضعیت غیر قابل تفکیک	State Variable Inserting
- در صورت وجود این دو خاصیت اجرای ترتیبی قوانین معادل اجرای موازی آنهاست	Check non-interfering and stability
- تولید قوانین مستقل بازای هر سیگنال	Generate Production Rule
- اعمال شرایط اولیه به مدار	Add Reset Signal
- دسته‌بندی عملوندهای دو قانون و تغییر (ضعیف یا قوی کردن)شرایط اجرایی آنها و تبدیل به اپراتورهای استاندارد C-element یا or-gate	Symmetrization
-کنترل عدم وجود تأخیرهای نا مساوی در مسیرهای انشعاب	Test Isochronic Fork Condition
- استفاده از متغیر مثبت در تغییر رو به پایین ومنفی در تغییر رو به بالا	Chang to CMOS Mappable
-بررسی زمان تغییر خروجی متناسب با برقراری مسیر از VDD به GND	Transistor Sizing & Optimization
- محاسبه اندازه مناسب ترانزیستور برای عملکرد بهینه شبکه ترانزیستورها	Fabrication

منجر به تولید تغییر ناخواسته شود. از جمله ساختارهای مورد استفاده در این روش انتساب مقدار، انتخاب در اجرا، تکرار، ترتیب عملیات، اجرای موازی پروسسها، ارتباطات بین پروسس و همزمانی پروسسها می‌باشد که در شکل ۴ قبلاً نمایش داده شده است.

از جمله پردازنده غیرهمگام طراحی شده با این روش نسخه غیرهمگام پردازنده MIPS R3000, ARM در دانشگاه Caltech را می‌توان اشاره کرد که دارای کارایی فوق‌العاده در بین تمامی نمونه‌های طراحی بوده و اولین نسخه تولید آن بدون مشکلی عملیاتی شده است. از مشخصات این پردازنده کارائی ۲۸۰MIPS در ولتاژ ۳/۳ ولت و دمای ۷۵ درجه و توان مصرفی ۷ وات یا توان مصرفی ۷۵٪ وات درولتاژ کاری ۱/۵ ولت با کارائی MIPS ۱۰۰ می‌باشد. لازم بیاد آوری است این پرداز نده در محدوده ولتاژ ۳٪ تا ۷ ولت می‌تواند عملیاتی باشد [۱۵].

د- STG

در این روش مشخصات مدار بکمک یک گراف جهت‌دار شبیه Petri Net شامل یکسری گره یا مکان و یکسری تغییرات بیان می‌شود [۱۱]. به این معنی که آتش کردن هرسیگنال بمعنای تغییر در سیگنال یا سیم مر بوطه خواهد بود. در این روش عملیات لازم جهت اتوماتیک کردن پروسسهای طراحی و حذف پیچیدگی نهایی مدار بر مبنای مدل Petri net ارائه میشود. یک نمونه STG و معادل Petri Net آن و گراف حالت مدار در شکل ۸ رسم شده است.



شکل ۸- نمونه STG و معادل Petri Net آن [۲۴]

برای تولید مدار از STG لازم است یکسری محدودیتهای روی STG به شرح زیر بررسی و کنترل شود:

- زنده‌بودن گرههای گراف به این معنا که بتوان ازهر مکانی هر تغییری را اجراکرد.
- قبل از تغییر مجدد ورودی، نباید خروجی هیچ گرهی تغییر کرده باشد.
- هیچ یک از مسیرها نباید دارای بیش از یک نشان باشند.
- تغییرات سیگنال بصورت مرتب یک و صفر و یا X^+ , X^- عوض شود.
- هیچ دو گره STG مقادیر یکسانی برای سیگنالها نداشته باشند.

موارد ۲و۳ برای تضمین عاری از تغییر ناخواسته بودن خروجی لازم بوده و بقیه موارد برای طراحی الگوریتمهای تولید مدار حائز اهمیت می‌باشد. یکی از روشهای پیاده‌سازی STG معروف به Contraction است که در آن برای تولید یک سیگنال تمامی حالتهای غیروابسته به یک خروجی (مرده) از STG حذف می‌شود. در واقع با شکستن STG به مدلهای کوچکتر متناظر با هر خروجی، پیچیدگی طراحی کاسته خواهد شد. نمونه‌ای از مدارهائی که STG آنها شرایط فوق را برآورده نمی کند در شکل ۹ آورده شده است.

در قسمت a گره r+ ممکن است مجدداً فعال نشود، بنابراین زنده نبودن همه گره‌ها در همه زمانها را ممکن می‌سازد. در مورد b پس از مراحل S+→r-→S-→ r+ دو فعالیت r+ و y+ دارای ۲ نشان خواهند بود. در مورد c بدلیل امکان فعال شدن x- قبل از y+ در y+→x+ شرط ۴ را برآورده نمی‌کند. در حالت d امکان تخصیص کد حالت سازگار بدلیل توالی دو تغییر X+ در مدار وجود ندارد. و در آخر در مدار e بدلیل پائین بودن سطح سیگنالها در دو حالت اولیه و x+→ y-، امکان کدگذاری سازگار به حالتهای مدار وجود ندارد.

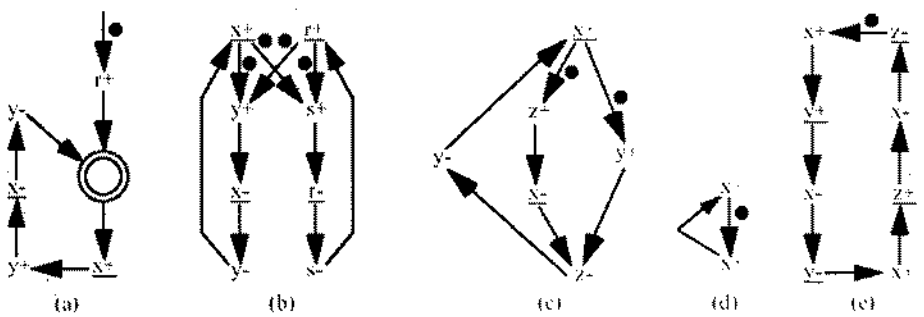
روش‌های SI مبتنی بر STG در طراحی Amulet تبلور یافته‌اند. [۱۷، ۱۸ و ۲۶]

۵- پردازنده های ناهمگام

اولین پردازنده غیرهمگام در سال ۱۹۸۸ بنام CAP توسط گروه مارتین در دانشگاه Caltech بمنظور کسب تجربه طراحی مدار براساس برنامه‌نویسی همزمان و تبدیلات متوالی آنها طراحی و در اولین نسخه تولید بطور کامل عملیاتی گردید [۲]. طراحی این پردازنده در ابتدا در ۸ برنامه موازی برای توابع مستقل بود که پس از تبدیلات متوالی به مدار الکترونیک تبدیل و با اتصال به یکدیگر در تکنولوژیهای ۱/۶ و ۲ میکرومستر MOSIS تولید گردید. کارائی این پردازنده در تکنولوژی ۲ میکرو MIPS ۱۲ و در ۱/۶ میکرو MIPS ۱۸ در ولتاژ ۳/۳ ولت بود. نسبت توان مصرفی به کارائی این پردازنده از ۶۰ MIPS/W برای تکنولوژی ۱/۶ میکرو و ولتاژ ۵ ولت تا ۶۰۰ MIPS/W برای تکنولوژی ۲ میکرومتر و ولتاژ ۲ ولت با زیزپردازنده آلفا با MIPS ۳۰۰ و توان مصرفی W ۳۰ و ولتاژ V ۳/۳ معادل ۱۰ MIPS/W قابل مقایسه میباشد.

پردازنده FAM در اوایل سال ۱۹۹۰ توسط CHO در دانشگاه TOKYO طراحی شده است. این پردازنده نیز شبیه CAP آزمایشگاهی بوده و با وجود مسیر داده ۳۲ بیتی و ۳۲ عدد ثبات ۳۲ بیتی فقط ۱۸ دستور پیاده‌سازی شده است. در این پردازنده برای همگام‌سازی ۲ واقعه از دو ترانزیستور در مسیر SET و RESET فیلیپ‌فلاپ‌ها (در آن یکی قابل کنترل توسط ورودی و یکی توسط فیلیپ‌فلاپ) استفاده شده است که کارائی بهتری از گیتهای استاندارد همگام‌سازی (C-element) را ارائه داده است. کارائی تخمینی این پردازنده MIPS ۳۰۰ محاسبه شده است.

پردازنده NSR در دانشگاه Utah در سال ۱۹۹۳ بر اساس ۵ عدد بلوک موازی (مطابق طبقات خط لوله پردازنده مشابه سنکرون آن) و خود همگام که از تعدادی بافر برای جلوگیری از توقف بین بلوکها استفاده شده بود، طراحی گردید. نسخه پیاده سازی شده FPGA این پردازنده کارائی MIPS ۱/۳ را ارائه داده است.



شکل ۹- نمونه هایی از مدارهای STG [۲۴]

جدول۱- مقایسه پردازنده های آسنکرون

نام پردازنده	سال ساخت	دانشگاه	تکنولوژی ساخت	ولتاژ کاری	کارائی MIPS
CAP	1988	Caltech	1.6 um	33 v	18
CAP	1988	Caltech	2 um	2 v	12
ALFA		Digital	0.5	3.3 v	300
FAM	1990	Tokyo			300
NSR	1993	Utah	FDGA		1.3
CFDD	1994	SUN			
STRIP		Stanford	2 um		62.5
Amulet	1997	Manchest	1 um		38

جدول ۲- مقایسه توان پردازش پردازنده های سنکرون

پردازنده	تکنولوژی ساخت	MIPS	Pw	MIP 2/E
Minimips 3.3v	0.6	280	7	3100
2 v Minimips	0.6	150	1	3375
Amulet 2	0.5	38	0.150	365
orion R46000	6.64	150	3.0	1125
ARMSALLO 2v	0.35	235	0.9	14420

پردازنده CFPP در سال ۱۹۹۴ بوسیله گروه SUTHERLAND در شرکت SUN طراحی و پیاده سازی شده است. ایده اصلی طراحی این پرداز نده استفاده از تکنیک خط لوله در تمامی قسمتهای کنترل مدار و استفاده از طبقات خالی برای جبران اختلاف تأخیر بین طبقات بوده است. پرداز نده STRIP بر مبنای مدل پردازنده همگام MIPS-X ولی با پالس ساعت قابل تنظیم در دانشگاه STANFORD طراحی شده است.

۶- جمع بندی و نتیجه گیری

همانطوریکه اشاره گردید طراحی ناهمگام دارای ابعاد غنی جهت تحقیقات و فن‌آوری می‌باشد. در حقیقت در این مقاله هدف ارائه جزئیات روشهای طراحی این مدارها نبوده و بلکه بیشتر هدف ارائه شناخت اولیه و شناساندن ابعاد وسیع و قابل گسترش این معماری بوده است. امیدواریم در آینده بتوان با ارائه پروژه‌های کاربردی و جدید، گامی در جهت پیاده‌سازی اهداف و ویژگیهای قابل انتظار این معماری برداریم.

مراجع

[1] A. J. Martin, "The Design of an Asynchronous MIPS R3000 Microprocessor," *PKO. IEEE,* 1997.

[2] A. J. Martin, S. M. Burns, T. K Lee, D. Borkovic, and P. J. Hazewindus, "The Design of an Asynchronous Microprocessor," in C.L.Sietz, editor, Advanced Research in VLSI: *Proceeding of the Decennial Caltech Conference on VLSI*, pp. 351-373, MIT Press, 1989.

[3] A. J. Martin, "The Limitation to Delay-Insesitivity in Asynchronous Circuits," *Sixth MIT Conference on Advanced Research in VLSI*, ed W. J. Dally, pp. 263-278, MIT Press, 1990.

[4] A. J. Martin, "Synthesis of Asynchronous VLSI Circuits," *Technical Report CS-TR-93-28,* Computer Science Department, California Institute of Technology, 1991.

[5] A. J. Martin, "Asynchronous Datapaths and the Design of an Asynchronous adder," *Formal Methods in System Design*, Vol. 1, No. 1, pp. 117-137, 1992.

[6] A. J. Martin, "Tommorow's Digital Hardware will be Asynchronous and Verified," In J. van Leeuwn, editor, *Algorithms, Software, Architecture, Information Processing* 92, Vol. I, Elsevier, 1992.

[7] J. T. Udding, "A Formal Model for Defining and Classifiying Delay-insensitive Circuits and Systems," *J. Didtributed Computing*, 1, 1986.

[8] J. C. Ebergen, "Translating Programs into Delay-Insensitive Circuits," Ph.D. Thesis, Technische Universiteit indhoven, 1987.

[9] J. C. Ebergen, "A Formal Approach to Designing Delay-insensitive Circuits", *Destributed Computing*, Vol. 5, No. 3, 1991.

[10] A. M. Cummings, A. M. Lines, A. J. Martin "An Asynchronous Pipelined Lattice Structure Filter," *Advanced Research in Asynchronous Circuits and system*, pp. 128-133, 1994.

[11] T. Murata, "Petri Nets: Properties, Analysis and Application," *Proc IEEE*, Vol. 77, No. 4, 1989.

[12] T. H. Y. Meng, R. W. Brodersen, and D. G. Messerschmitt, "Automatic Synthesis of Asynchronous Circuits form High-level Specifications," *IEEE Trans. Comp-Aided Design*, Vol. 8, 1989.

[13] T. A. Chu, "Synthesis of Self-Timed VLSI Circuits from Graph-Theoric Specifications," Ph.D. Thesis, Massachusetts Institute of Technology, 1987.

[14] T. W. Venkatech, "Asynchronous Processor Survey," University of California Technical paper 0018-9162/97, IEEE, 1997.

[15] P. A. Beerel, K. Y. Yun, "The Design and Verification of A High-Performance Low-Control_Overhead Asynchronous Differential Equation Solver," *IEEE Trans. On VLSI System*, 1997.

[16] P. G. Lucassen and J. Udding, *"A Process-Algebraic Approach to the Design of Asynchronous Pipelines,"* IEE, 1998.

[17] A. Yakovlev and A. Semenov, "Synthesis of Speed Independent Circuits from STG-unfolding Segment," Technical Paper GR/J 52327, University of Newcastle, 1998.

[18] A. Yakovlev and A. Kondratyev, "Technology Mapping for Speed-Independent Circuits: Decomposition and Resynthesis," Technical Paper GR/L 24038, University of Newcastle, 1997.

[19] L. G. Birtwistle, "Modelling Amulet3 in LARD," Amulet Project Technical Paper, Manchester Univaersity, 1998.

[20] A. M. Lines, "Pipelined Asynchronous Circuits," Caltech Technical Paper, 1998.

[21] A. J. Martin and R. Manohar, "Quasi-delay-insensitive Circuits are Turing-complete," Caltech Tachnical Paper CS-TR-95-11, 1995.

[22] S. B. Endecott and S. B. Furber, "Modelling and Simulation of Asynchronous System using the LARD Hardware Description Language," *Proceeding of the 12th European Simulation Multiconference*, 1998.

[23] D. A. Bardsley and D. A. Edwards, "Compiling the Language Balsa to Delay Insensitive Hardware," *Proceedings CHDL'97*, Toledo, 1997.

[24] S. Hauck, "Asynchronous Design Methodologies: An Overview," *Proceeding of IEEE*, Vol. 83, No. 1, 1995.

[25] "Caltech Asynchronous VLSI Design Current Research", *Caltech Journal*, 2000.

[26] "AMULET Group Theses," *Journal of University of Manchester*, 2000.

[27] J. R. Burch, "Modeling Timing Assumption with Trace Theory," *IEEE Transaction on Computers*, 1998.

[28] C. J. Myers, "Technology Mapping of Timed Circuits," *IEEE Transaction on Design and Test*, 1995.

[29] P. G. Lucassen, "Normal Form in DI-Algebra with Recursion," *IEEE Transaction on Computer*, 1997.

[30] R. Negulescu, "Relative Liveness: From Intuition to Automated Verification," *IEEE Design and Test*, 1995.

[31] T. J. Chaney, C. F. Molnar, "Anomalous Behavior of Synchroniyers and Arbiters," *IEEE Transactions on Computers*, Vol. C-22, pp. 421-422, 1973.

[32] C. Mead and L. Conwany, *Introduction to VLSI System*, Addison-Wesley, 1980.

[33] S. H. Unger, *Asynchronous Sequential Switching Cicuits*, Wiley-Interscience, 1969.

[34] S. M. Norwick and D. L. Dill, "Exact Two-level Minimization of Hazard-Free Logic With Multiple-Input Changes," *Proceedings of ICCAD*, pp. 629-630, 1992.

[35] I. E. Sutherland, "Micropipelines," Communications of the ACM, Vol. 32, No. 6, pp. 720-738, 1989.

سیدولی ا... فاطمی کارشناسی و کارشناسی ارشد در مهندسی کامپیوتر را در سال های ۱۳۶۸ و ۱۳۷۱ از دانشگاه صنعتی شریف اخذ نمود. در حال حاضر دانشجوی دوره دکترای مهندسی کامپیوتر در دانشگاه صنعتی امیرکبیر می‌باشد. موارد تحقیقاتی مورد علاقه ایشان پردازش موازی، طراحی ابزار طراحی سخت‌افزار و طراحی سیستم‌های آسنکرون می‌باشد. ضمناً ایشان دارای تجربیات ممتد در طراحی و پیاده‌سازی و **اجرای** سیستم‌های بانکی مدرن می‌باشد.

حسین پدرام در سال ۱۳۵۶ مدرک کارشناسی مهندسی برق خود را از دانشگاه صنعتی شریف و در سال ۱۳۵۹ مدرک کارشناسی ارشد مهندسی برق خود را از دانشگاه ایالتی اوهایو در امریکا دریافت نمود. از سال ۱۳۶۱ الی ۱۳۶۵ نامبرده به عنوان کارشناس ارشد سیستم های دیجیتال در مرکز تحقیقات مخابرات ایران به کار مشغول بود و پس از آن به دوره دکترای مهندسی برق و کامپیوتر در دانشگاه ایالتی واشنگتن در سیاتل امریکا وارد گردید و در سال ۱۳۷۰ موفق به اخذ درجه دکترا در مهندسی کامپیوتر از دانشگاه مذکور گردید. دکتر پدرام از سال ۱۳۷۱ در دانشکده مهندسی کامپیوتر دانشگاه صنعتی امیرکبیر در تهران مشغول به فعالیت گردید و اینک نیز عضو هیأت علمی این دانشکده می‌باشد. زمینه های علمی مورد علاقه نامبرده متنوع بوده و شامل موضوعاتی مانند ایده‌های نو در معماری کامپیوتر، مدیریت شبکه‌های کامپیوتری، سیستم‌های توزیعی و رباتیک می‌باشد. آدرس پست الکترونیکی نامبرده عبارتست از: pedram@ce.aut.ac.ir