

الگوریتم جدید جذر با استفاده از سیستم‌های Multiple Expert

علی ذاکر الحسینی*

سمیه تیمارچی*

چکیده

اساساً الگوریتم‌های جذر را می‌توان به دو مجموعه اصلی تقسیم کرد: روش‌های «تکراری» و روش‌های «تقریبی با توابع حقیقی». الگوریتم‌های تکراری راه حل‌های کندر اما دقیق‌تری را نسبت به الگوریتم‌های تقریبی پیشنهاد می‌کنند. یکی از عیبهای روش‌های تقریبی استفاده از LUT برای بدست آوردن یک تقریب اولیه می‌باشد. از آنجا که سربار سخت افزاری این جداول بسیار زیاد است، در این مقاله، راه حل‌هایی برای بدست آوردن تقریب اولیه پیشنهاد شده است. در این زمینه دو روش ارائه می‌شود که با استفاده از Hybrid Multiple Expert System می‌شوند.

کلمات کلیدی

.VHDL، Hybrid Multiple Expert System، روش‌های تکراری، روش‌های تقریبی با توابع حقیقی

A Novel Square Root Algorithm Using Multiple Expert Systems

Abstract

The square root algorithms are mainly based on “iteration” and “approximation with real functions” methods. The iteration methods suggest slower but more accurate methods than the approximation. One of the main drawbacks of approximation methods is the selection of first approximation using look up tables (LUT). As the hardware overhead of using such LUT is great, this paper suggest solutions for estimating the first approximation. Two methods using “Hybrid Multiple Expert” for increasing the performance of square root algorithms are presented.

Keywords

Square root algorithm, Hybrid Multiple Expert System, VHDL.

* دانشجوی دکترای معماری سیستم‌های کامپیوتری دانشگاه شهید بهشتی، دانشکده مهندسی برق و کامپیوتر، s_timarchi@sbu.ac.ir
† عضو هیأت علمی دانشگاه شهید بهشتی، دانشکده مهندسی برق و کامپیوتر، a_zaker@sbu.ac.ir

۱- مقدمه

تابع جذر یکی از مهمترین توابع اولیه می‌باشد که در کاربردهای سیاری مورد استفاده قرار می‌گیرد [11]. الگوریتم‌های جذر عموماً به دو مجموعه اصلی، روش‌های تکراری و روش‌های تقریبی با تابع حقیقی تقسیم می‌شوند. روش‌های تکراری را می‌توان به ۳ دسته روش‌های مستقیم، الگوریتم‌های مبتنی بر فرمول نیوتون-رافسون و روش‌های نرمال‌سازی تقسیم‌بندی نمود.

روش‌های مستقیم و برخی تکنیک‌های نرمال‌سازی، بیشتر مربوط به پیاده‌سازی‌های سخت افزاری می‌باشد. دیگر روش‌های تکراری و تقریب توسط تابع حقیقی را می‌توان هم به صورت سخت افزاری و هم نرم افزاری پیاده‌سازی کرد [10]. روش‌های قلم و کاغذ، استردادی^۱ و غیر استردادی^۲ (NR) [5, 6] به طبقه روش‌های مستقیم مربوط می‌شود. الگوریتم‌های CORDIC [4] Delugish's، Chen's [4] روش‌های مشتق از نیوتون-رافسون هستند [10]. از میان روش‌های تکراری جذر، روش‌های NR و CORDIC در این مقاله پیاده‌سازی خواهند شد.

تقریب توسط تابع حقیقی، راه دیگر برای مسأله جذر می‌باشد. بهترین تقریب‌های شناخته شده توسط تابع حقیقی عبارتند از: سری تیلور، مک لوران و چند جمله‌ای ساده چبیچف. چون استاندارد IEEE 754 ممیز شناور نیاز به نتیجه صحیح در بیت آخر دارد، که به سادگی توسط ارزیابی‌های چند جمله‌ای ساده به دست نمی‌آید، تقریب با تابع حقیقی همچنان در تئوری ریاضیات بجز برای یک تعداد محدود از کاربردهای خاص، بسیار مورد استفاده قرار گرفته است [10]. در این مقاله، روش‌های تکه‌ای^۳ و درونیابی خطی^۴ از این دسته مورد ارزیابی قرار خواهند گرفت.

تراشه‌های FPGA به شکل گسترهای در کاربردهای مربوط به ASIC و پیاده‌سازی مدار کنترل بکار می‌رود [14]. انعطاف‌پذیری FPGA امکان توسعه سریع ساخت افزارهای با کارایی بسیار بالا را می‌دهد [13, 2]. چند ویژگی در FPGA وجود دارد که هنگام انتخاب و پیاده‌سازی الگوریتم‌های ریاضی برای محاسبه عملیات ریاضی خاص، بسیار مفید می‌باشد. تراشه FPGA دارای بلوکهای حافظه داخلی است که می‌تواند به شکل کارآمدی جهت پیاده‌سازی LUT بکار رود. مزیت بسیار مهم دیگر قطعات منطقی قابل برنامه‌سازی این است که به شکل ذاتی برای یک کلمه با طول خاص محدود نمی‌باشد. به عنوان مثال چنانچه عملی نیاز به دقت ۱۱ بیت داشته باشد، تمام عملیات ریاضی می‌تواند دقیقاً با دقت ۱۱ بیت انجام شود [14].

در این مقاله از اکسپرتهای^۵ زیر برای محاسبه جذر استفاده شده است:

- ۱- الگوریتم NR دودویی
- ۲- الگوریتم CORDIC
- ۳- الگوریتم تکه‌ای

هدف این است که با ترکیب اکسپرتهای اشاره شده و تشکیل یک سیستم که در آن از دو یا چند نوع الگوریتم با ترتیب خاص استفاده شده است، یک سیستم Multiple expert بسته آید که بتواند کارایی محاسبه جذر را نسبت به روش‌های مجرد بهبود بخشد. این مقاله شامل معرفی بر الگوریتم‌های محاسبه جذر، ارائه چند سیستم اکسپرت جهت محاسبه جذر، بررسی مزیت‌ها، محدودیت‌ها و کاربردهای هر روش با توجه ویژه به پیاده‌سازی سخت افزاری آنها، ارائه شده است. سیستم‌های پیشنهادی با روش‌های موجود مقایسه شده و نتایج شبیه‌سازی بر روی FPGA ذکر خواهد شد.

۲- توصیف الگوریتم‌های متداول جذر

۲-۱- الگوریتم CORDIC

این الگوریتم یک راه حل تکرار شونده^۶ برای تابع مثلثاتی و غیرتجربی است. تابع مثلثاتی مبتنی بر دوران‌های برداری هستند؛ در حالیکه دیگر تابع مانند جذر با استفاده از یک عبارت افزایشی از تابع مطلوب، پیاده‌سازی می‌شود. این الگوریتم CORDIC نام دارد که مخفف Coordinate Rotation Digital Computer می‌باشد. در این الگوریتم، تنها از شیفت و جمع برای محاسبات استفاده می‌شود. اصطلاحاً گفته می‌شود این الگوریتم مبتنی بر شیفت و جمع می‌باشد [7, 8, 9, 11, 12, 15].

توابعی که با استفاده از CORDIC قابل محاسبه هستند، سینوس، کسینوس، تانژانت، کوتانژانت، تابع هاپرボلیک^۷، لگاریتم طبیعی، تابع نمایی، ریشه دوم، ضرب، تقسیم به صورت دودویی. معادلات بازگشته که تابع مذکور توسط آن، ارزیابی می‌شوند، به صورت ذیل می‌باشند:

$$X_{k+1} = X_k - m d_k Y_k 2^{-k}$$

$$Y_{k+1} = Y_k - m d_k X_k 2^{-k}$$

$$Z_{k+1} = Z_k - d_k e_k$$

که $\{e_k\}_{k=0}^n \in \{-1, 0, 1\}$ نشانده‌نده «حالت» می‌باشد.

$\{d_k\}_{k=0}^n$ رشته‌ای از ثوابت از قبل محاسبه شده وابسته به m است و $\{d_k\}_{k=0}^n \in \{-1, 0, 1\}$ و نیز مقادیر اولیه X_0, Y_0, Z_0 به شکل مناسبی انتخاب می‌شود. مقادیر اولیه لازم برای بدست آوردن تابع فوق در مرجع [11] ذکر شده است.

در پیاده‌سازی CORDIC به ۳ عدد ثبات Z , y , x ، یک lookup table (LUT) برای ذخیره مقادیر e_k و دو شیفت‌دهنده برای محاسبه $x 2^{-k}$ و $y 2^{-k}$ به عنوان ورودی دوم واحدهای جمع/تفريق کننده نیاز است [18]. ضریب d_k که متعلق به مجموعه $\{-1, 0, 1\}$ است با انتخاب عملوند یا مکملش انجام می‌شود [11].

۲-۳- الگوریتم درونیابی خطی

یکی از روش‌های ممکن برای پیاده‌سازی جذر، ذخیره حاصل جذر تمام اعداد در جدول می‌باشد. این راه حل نیاز به حافظه بسیار بزرگی دارد زیرا اندازه مدار، وابستگی مستقیم به اندازه و دقت عدد دارد. به عنوان مثال، اگر رشته از ۴ بیلیون عدد تشکیل شده باشد و ۴ بایت برای ذخیره هر عدد، فضای نیاز باشد، تراشه باید به اندازه ۱۶ گیگا بایت حافظه داشته باشد که این مقدار برای بزرگترین FPGA های تجاری موجود نیز مناسب نیست. یکی از روش‌های کارآمد و انعطاف پذیر برای تولید این اعداد، ترکیب روش نرم افزاری با یک جدول است [10, 11]. این روش به این صورت است که:

چنانچه مقدار تابع $f(x)$ برای $x = X_{high}$ و $x = X_{low}$ که در آن $X_{low} < X_{high}$ است مشخص باشد، مقدار تابع برای x که در فاصله $[X_{low}, X_{high}]$ است، از $f(X_{low})$ و $f(X_{high})$ توسط روش درونیابی خطی یا Linear Interpolation (LI) قابل محاسبه است. در روش درونیابی خطی، $f(x)$ برای x های در بازه $[X_{low}, X_{high}]$ توسط فرمول زیر قابل محاسبه است:

$$f(x) = f(X_{low}) + \frac{(x - X_{low})(f(X_{low}) - f(X_{high}))}{(X_{high} - X_{low})} \quad (1)$$

از زیبایی چنین عبارتی، ۴ عمل جمع و یک عمل ضرب و یک عمل تقسیم نیاز دارد. اما با انتخاب نقاط انتهایی X_{low} و X_{high} که توانهایی از دو باشند، تقسیم و دو تا از عمل‌های جمع به عملیات جزئی تری کاهش می‌یابد [11].

فرض کنیم X یک عدد n بیتی باشد. در زیر، لیست مراحل این روش آمده است [18]:

۱. بیت بالایی X را R نامگذاری می‌کنیم.
۲. بیتهای باقیمانده X را S فرض می‌کنیم.
۳. با دادن R به عنوان ورودی آدرس جدول LUT، دو مقدار متواالی از جدول را خوانده، آنها را A و B نامگذاری می‌کنیم (A و B در واقع همان $f(X_{low})$ و $f(X_{high})$ می‌باشند).
۴. توسط رابطه (۱) و بکار بردن S به عنوان آفست دو عدد، یک نقطه میان A و B بدست می‌آوریم. در واقع رابطه (۱) به صورت زیر می‌شود:

$$f(x) = A + \frac{Z(B - A)}{2^{N-k}}$$

که در آن Z یک عدد n بیتی است که k بیت بالایی آن صفر و $N-k$ بیت پایینی اش برابر S است و $f(x)$ تابع چگالی احتمال توزیع نمایی است. با توجه به رابطه فوق تقسیم به یک عمل شیفت تبدیل می‌شود:

$$f(x) = Z[(B - A) \gg (N - k)] + A$$

انتخاب عمل جمع یا تفرقی که با d_i مشخص می‌گردد توسط علامت ثبات‌های y یا Z و با توجه به اینکه در حالت برداری یا دورانی قرار دارد، تعیین می‌گردد. در عمل، مقادیر اولیه از طریق مالتی‌پلکسر به داخل ثبات‌های X , Y , Z ریخته می‌شود. سپس در هر یک از n سیکل بعدی ساعت، مقادیر ثبات‌ها از طریق شیفت‌دهنده‌ها و جمع/تفرقی کننده‌ها در هر تکرار تغییر می‌کنند تا ثبات‌ها ذخیره می‌شود. شیفت‌دهنده‌ها در هر تکرار آدرس LUT نیز در هر تکرار، افزایش می‌یابد. بنابراین مقدار زاویه اولیه مناسب وارد واحد جمع/تفرقی کننده Z می‌شود [1].

۲-۲- الگوریتم تکه‌ای

یکی از روش‌های محاسبه توابع مقدماتی، روش‌های مبتنی بر جدول (Lookup Table) با استفاده از بخش‌هایی از عملوندهای ورودی Piecewise Lookup Table (PLT) می‌باشد. الگوریتم تکه‌ای یا n بیتی X را به ۴ بخش تقسیم می‌کند [11]

$$\begin{aligned} x &= t + \lambda u + \lambda^2 v + \lambda^3 w \\ \Rightarrow x &= t + 2^{-6} u + 2^{-12} v + 2^{-18} w \end{aligned}$$

برای $n=32$ ، هر کدام از مؤلفه‌های t, u, v, w و t یک عدد ۸ بیتی هستند. چندجمله‌ای تیلور برای $f(x)$ به صورت ذیل می‌باشد:

$$f(x) = \sum_{i=0}^{\infty} f^{(i)}(t + \lambda u) \frac{(\lambda^2 v + \lambda^3 w)^i}{i!}$$

با استفاده از چندجمله‌ای تیلور، می‌توان تقریب زیر را

برای $f(x)$ بدست آورد که دقت آن $O(\lambda^5)$ است:

$$f(x) \approx f(t + \lambda u)$$

$$+ \frac{\lambda}{2} [f(t + \lambda u + \lambda v) - f(t + \lambda u - \lambda v)]$$

$$+ \frac{\lambda^2}{2} [f(t + \lambda u + \lambda w) - f(t + \lambda u - \lambda w)]$$

$$+ \lambda^4 \left[\frac{v^2}{2} f^{(2)}(t) - \frac{v^3}{6} f^{(3)}(t) \right]$$

بدین ترتیب محاسبه $f(x)$ شامل عملیات زیر می‌باشد:

۱. چهار مقدار $t + \lambda u + \lambda w$, $t + \lambda u - \lambda v$, $t + \lambda u - \lambda w$ و $t + \lambda u - \lambda v$ نیاز به چهار عمل جمع دارند. ($t + \lambda u$ به محاسبه ندارد).
۲. خواندن ۵ مقدار $f(x)$ از یکی از جداول LUT

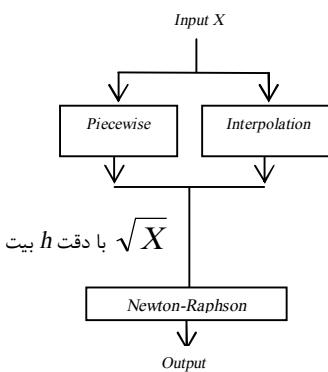
۳. خواندن آخرین عبارت $\frac{v^2}{2} f^{(2)}(t) - \frac{v^3}{6} f^{(3)}(t)$ که تابعی از v می‌باشد از یک جدول دیگر

۴. انجام یک عمل جمع ۶ عملوندی برای محاسبه نهایی

مدت زمان تعیین شده توسط سیستم، هر کدام از اکسپرت‌ها به نتیجه با دقت بیشتر رسیده باشد، آن خروجی انتخاب شود. در این حالت، خروجی‌ها علاوه بر مقدار جذر باید عددی را که نشان دهنده دقت بیت خروجی است را نیز شامل شوند.

۲-۳- سیستم‌های SRMES1 و SRMES2

ایده این روش، موازی کردن دو الگوریتم درونیابی خطی و تکه‌ای است (شکل ۱۱). سرعت روش درونیابی خطی بستگی به دقت خروجی دارد یعنی تقریباً h پالس ساعت لازم است تا یک خروجی با دقت h بیت ایجاد شود و از طرفی روش تکه‌ای برای تولید خروجی با دقت h بیت، تعداد پالس ساعت ثابتی که آنرا k فرض می‌کنیم لازم دارد. برای h های نسبتاً بزرگ، داریم $h > k$. برای همزمان کردن این دو اکسپرت که به صورت موازی می‌توان از آنها استفاده کرد، h را برابر k فرض کرده تا در k پالس ساعت، دو خروجی با دقت $h = k$ بیت تولید شود. در مرحله بعد، برای افزایش دقت جذر از میان روش‌های متداول جذر، تنها می‌توان از روش نیوتون-رافسون استفاده نمود. در این روش هرچه ورودی دقیق‌تر باشد، زودتر به خروجی نهایی می‌رسیم.



شکل (۱): ساختار یک HMES جهت محاسبه جذر

برای ساخت نهایی اولین سیستم ترکیبی پیشنهادی یعنی شکل (۲) را پیشنهاد می‌کنیم. این ساختار از موازی کردن ساختار شکل (۱) و روش‌های دیگر مثل NR، CORDIC یا دیگر روش‌های جذر بدست می‌آید. الگوریتم تصمیم‌گیری در این شکل، با توجه به کاربرد مورد نظر، خروجی نهایی را از میان خروجی‌های بدست آمده از مسیرهای مختلف تولید می‌نماید.

در دومین سیستم پیشنهادی یعنی Square Root Multiple Expert System 2 (SRMES2) می‌توان موازی با دو اکسپرت روش تقریبی در ساختار شکل (۱)، روش‌های دیگر را نیز بکار برد (شکل (۳)). لازم به ذکر است که ترکیب‌های متنوعی از الگوریتم‌ها را برای تولید جذر می‌توان بکار برد. در این مقاله تنها به ۲ نمونه از آنها اشاره گردید.

۳- ارائه چند سیستم برای Multiple Expert محاسبه جذر

یکی از عیوب‌های روش‌های تقیبی استفاده از LUT برای بدست آوردن یک تقریب اولیه می‌باشد. از آنجا که سربار سخت افزاری این جداول سیار زیاد است، یک راه حل این است که از یک روش جذر دیگر برای بدست آوردن تقریب اولیه استفاده شود. در این زمینه دو روش ارائه نموده‌ایم که در زیر بخش‌های ۳-۳ و ۴-۳ به توصیف آنها خواهیم پرداخت. این روش‌ها (HMES) Hybrid Multiple Expert System [16] هستند که از ترکیب چند روش با ترتیب خاص برای محاسبه جذر استفاده می‌کنند.

Rosh Diger برای ساخت (MES) Multiple Expert System (MES) جهت محاسبه جذر این است که روش‌هایی را که می‌توان از آنها به عنوان اکسپرت‌های میانی استفاده کرد تعیین نمود. یکی از این روش‌ها، روش نیوتون-رافسون است. روش‌های دیگر مثل NR و CORDIC با مقادیر اولیه مشخص شروع به اجرای الگوریتم می‌کنند. به عبارت دیگر فرض کنید حاصل جذر یک عدد را تا دقت l_1 بیت در دست داریم. دو الگوریتم مذکور، این قابلیت را ندارند که با دریافت این مقدار، دقت را به l_2 بیت برسانند. در نتیجه نمی‌توانند به عنوان اکسپرت‌های میانی بکار روند. در این مورد، دو سیستم MES در زیر بخش ۲-۳ پیشنهاد شده‌است که به آن می‌پردازیم. ضمناً هر سیستم MES در مرحله آخر نیاز به الگوریتم تصمیم‌گیری خاصی دارد.

۳-۱- الگوریتم تصمیم‌گیری

برای تعیین الگوریتم تصمیم‌گیری باید به کاربرد روش توجه نمود. در این قسمت ۳ کاربرد را ذکر خواهیم کرد:

۱. محاسبات تحمل پذیر خطأ

در کاربردهای با محاسبات تحمل پذیر خطأ، الگوریتم تصمیم‌گیری می‌تواند به این صورت باشد که از میان اعداد تولید شده، عددی را که اختلاف نسبی با بقیه دارد جدا نموده و از مابقی میانگین گرفته شود. این روش می‌تواند در مدارات حسابی بکار رود که نیاز به قابلیت اطمینان بالا دارد.

۲. کاربردهای با موجودیت^۱ بالا

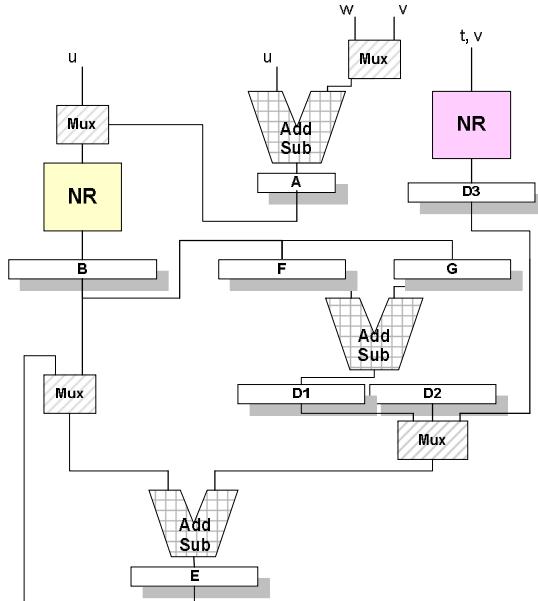
در یک سیستم، ممکن است از یک زمان t_0 به بعد بخش‌هایی از سیستم به دلیل نویز خارجی، خرابی قطعات و غیره از کار بیفتد. بنابراین وجود اکسپرت‌های دیگر، این امکان را می‌دهد که در لحظه t_0 ، کل مدار برای کاربر موجودیت داشته باشد. الگوریتم تصمیم‌گیری به این صورت عمل می‌کند که خروجی را از میان پاسخ‌های اکسپرت‌های سالم انتخاب نماید.

۳. کاربردهای بی‌درنگ^۲

از آنجا که در این سیستم‌ها سرعت مهمترین پارامتر است، الگوریتم تصمیم‌گیری را می‌توان به این صورت تعیین نمود که در یک

معادل تعداد تکرارهای لازم در روش NR می‌باشد. بنابراین کافی است در روش تکه‌ای به جای LUT از روش NR استفاده شود که در شکل (۴) مشخص گردیده است.

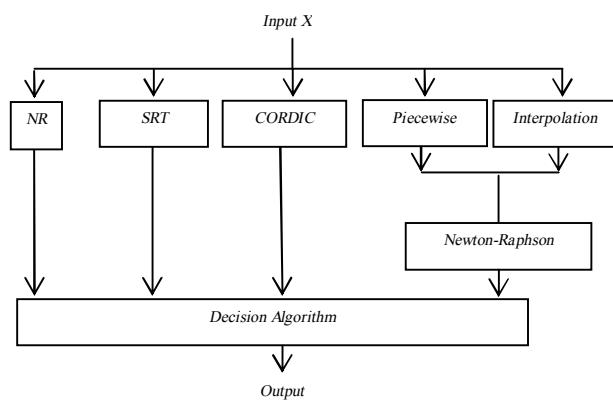
عیب این روش این است که روش NR برای محاسبه جذر ابتدا باید یک عمل تکراری که تعداد آن مناسب با طول دقت خروجی (h) است انجام دهد؛ در حالیکه زمان خواندن از LUT بسیار سریع می‌باشد. با توجه به اینکه برای ذخیره این جدول نیاز به سربار سخت‌افزاری بسیار زیادی داریم، با حذف LUT و استفاده از NR، سربار سخت‌افزاری بسیار کاهش خواهد یافت. در نتیجه با افزایش مقداری تأخیر بهبود زیادی در حجم سخت‌افزاری حاصل می‌گردد.



شکل (۴): ساختار NRP پیشنهادی

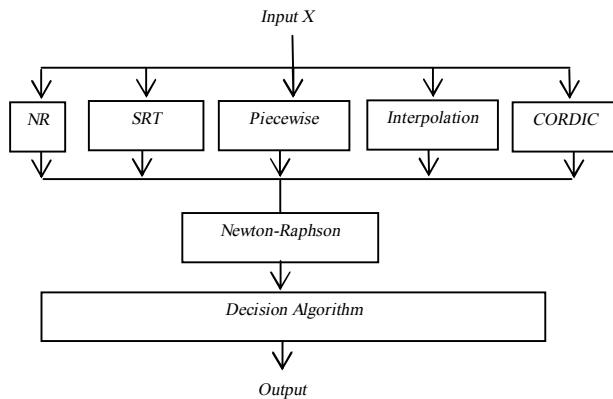
۴-۳- سیستم مرکب از دو روش میانگین‌گیری و غیر استردادی (NRLI)

روش درونیابی خطی (LI) یک روش تقریبی است که عملیات خود را با دو عدد که تقریب اولیه‌ای از جذر واقعی عدد ورودی است آغاز می‌کند. که این تقریب اولیه توسط یک LUT بدست می‌آید. با توجه به اندازه بسیار بزرگ این جدول روشی در این بخش ارائه خواهیم نمود که با استفاده از یک اکسپرت مانند NR تقریب اولیه را تولید نماید. ساختار اصلی این سیستم همان ساختار روش میانگین‌گیری است [18] که در آن از اکسپرت NR به جای LUT استفاده شده است. ساختار حاصل را NRI (Non-Restoring & Linear Interpolation) نامیده‌ایم که در شکل (۵) رسم شده است. لازم به ذکر است که علاوه بر اکسپرت NR می‌توان از اکسپرت‌هایی چون CORDIC، نیوتون-رافسون یا برخی روش‌های دیگر استفاده نمود. بنابراین می‌توان با انتخاب هر یک از این موارد، بهبود یا عدم بهبود کارایی را نسبت به MES های دیگر بررسی نمود.



شکل (۲): ساختار SRMES1 پیشنهادی

در SRMES1 سرعت کمتر از SRMES2 است چون مسیر NR و CORDIC در نهایت پاسخ‌های متعددی برای الگوریتم تصمیم‌گیری تولید می‌نماید که می‌تواند در ایجاد خروجی نهایی با معیارهای دلخواه مناسب‌تر باشد.



شکل (۳): ساختار SRMES2 پیشنهادی

۳-۳- سیستم مرکب از دو روش تکه‌ای و غیر استردادی (NRP)

همانطور که گفته شد، روش تکه‌ای (PLT) یک روش تقریبی است که عملیات خود را با چند عدد که تقریب اولیه‌ای از جذر واقعی عدد ورودی است آغاز می‌کند. در بخش ۲-۲ مشاهده شد که این تقریب اولیه توسط یک LUT بدست می‌آید. اما از نتایج شبیه‌سازی مشاهده می‌شود که اندازه این جدول از نظر عرض (میزان دقت تقریب اولیه) و طول، بسیار بزرگ است. از این‌رو روشی در این بخش ارائه خواهیم نمود که با استفاده از یک اکسپرت جذر، تقریب اولیه تولید شود. روشی که برای این کار انتخاب گردیده است الگوریتم NR است. عملکرد این روش بدین صورت است که در هر تکرار، یک بیت حقیقی از جذر بدست می‌آید. بنابراین همانطور که عرض LUT در روش تکه‌ای NRP معادل دقت تقریب اولیه می‌باشد، در سیستم Restoring & Piecewise نیز تعداد بیت‌های صحیح تقریب اولیه با همان تعداد تکرار در روش NR بدست می‌آید. به عبارت دیگر عرض

تعیین می‌گردد. پارامترهای درونی شامل طول و ارتفاع جدول LUT و تعداد بیت‌های ضرب و جمع می‌باشد. اندازه و سرعت روش درونیابی خطی نسبت مستقیم با پهنای هر خانه از جدول (W) و ارتفاع آن دارد. واحدهای بکار رفته در الگوریتم‌ها، شامل جمع کننده، تفریق کننده، ثبات‌ها، مالتی پلکسراها و جداول LUT می‌باشند. در آزمایشات انجام شده جمع/تفریق کننده‌ها سریال بوده و در n پالس ساعت کار خود را به پایان می‌برند که n تعداد بیت‌های عدد ورودی می‌باشد. ساختار دو روش پیشنهادی NRP و NRLI نیز که مبتنی بر ۳ روش NR، PLT و LI هستند در شکل‌های (۴) و (۵) ارائه شدند. خروجی برنامه فوق در یک فایل به گونه‌ای ذخیره شده است که قالب یک فایل VHDL را داشته باشد. سپس می‌توان این مؤلفه را به مؤلفه‌های دیگر اضافه نمود و با یک معماری که بدنۀ اصلی الگوریتم‌ها را تشکیل می‌دهد آنها را به هم مرتبط کرد.

۲-۴- نتایج مقایسه سرعت و سربار سخت‌افزاری

در شبیه‌سازی‌های انجام شده، طول ورودی و خروجی در تمام الگوریتم‌ها، ثابت و برابر ۳۲ بیت می‌باشد. همانطور که در جدول ۱ نشان داده شده است روش CORDIC دارای کمترین سرعت می‌باشد و بهترین سرعت را روش تکه‌ای دارد.

جدول ۱ نتایج حاصل از حجم مدارات سنتز شده در FPGA نوع Xilinx Spartan3 را نیز نشان می‌دهد. سنتز این مدارات توسط نرم‌افزارهای Leonardo و ISE Foundation انجام گرفته است. نتایج نشان می‌دهد که دو مدار پیشنهادی NRP و NRLI به ترتیب دارای کمترین سربار سخت‌افزاری می‌باشند. حجم این مدارات بسیار کمتر از روش‌های مجرد ذکر شده در جدول است.

جدول(۱): نتایج مقایسه سرعت و سربار سخت‌افزاری

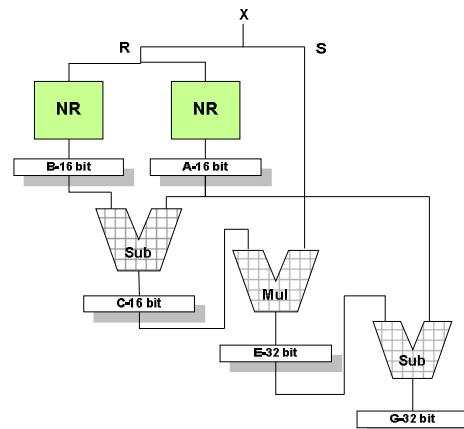
Method	Number of CLBs	The Method Frequency
CORDIC	2027	692.8 ns
PLT	9600	49.86 ns
LI	64000	83.657 ns
NRLI	345	213.1815 ns
NRP	294	179.3845 ns

بهبود کارایی NRLI نسبت به درونیابی خطی و نیز بهبود NRP نسبت به تکه‌ای با در نظر گرفتن معیار $\text{area} \times \text{time}^2$ با توجه به معادلات جدول ۲ بدست می‌آید:

با معیار $\text{area} \times \text{time}^2$ بهبود کارایی روش NRLI نسبت به درونیابی خطی (LI) برابر است با:

$$\frac{447903593.536}{15678991.42} \approx 28.567$$

و بهبود کارایی NRP نسبت به روش تکه‌ای (PLT) برابر است با:



شکل (۵): ساختار NRLI پیشنهادی

۴- نتایج ارزیابی مدارات جذر پیشنهادی

۴-۱- سیستم‌های شبیه‌سازی شده

در این بخش نتایج حاصل از شبیه‌سازی و سنتز تابع جذر با استفاده از روش‌های NR، CORDIC، تکه‌ای و درونیابی خطی و نیز سیستم‌های پیشنهادی NRP و NRLI ذکر خواهد شد. روش‌های پیاده‌سازی شده را با معیارهایی چون سرعت و سربار سخت‌افزاری با یکدیگر مقایسه خواهیم کرد.

در روش CORDIC، ساخت مولد جذر، نیاز به یک مدار جذر و یک مدار تقسیم کننده دارد. در نتیجه مجموعه‌ای از دو مدار، یعنی مدار جذر گیرنده و تقسیم کننده CORDIC باید پیاده‌سازی شود. ساختار اصلی این الگوریتم در شکل (۲) نشان داده شد. مدار جذر به تمام مولفه‌های نمایش داده شده در آن شکل نیاز دارد، اما مدار تقسیم تنها به ۳ ثبات، ۲ واحد جمع/تفریق کننده، یک شیفت دهنده و یک جدول کوچک نیاز دارد. در نتیجه جماعت ۶ ثبات، دو جدول (LUT)، ۵ شیفت دهنده و ۵ واحد جمع/تفریق کننده لازم است.

طرح CORDIC نیاز به دو پارامتر دارد: تعداد بیت‌ها در هر مرحله تکرار و تعداد مراحل انجام کار. در حالت کلی، CORDIC یک بیت در هر مرحله به نتیجه نهایی همگرا می‌شود. شیفت دهنده‌ها در هر بار تکرار، لازم است یک شیفت انجام دهند؛ در نتیجه در یک پالس ساعت کار خود را به پایان خواهند رساند.

در ساختار PLT (Piecewise Lookup Table) پیاده‌سازی شده، ۳ واحد جمع/تفریق کننده، ۸ ثبات با طول‌های مختلف، ۳ مالتی پلکسرا و دو جدول LUT به صورت خط لوله‌ای^{۱۱} با هم کار می‌کنند. برای پیاده‌سازی روش درونیابی خطی (LI)، یک تفریق کننده، یک ضرب کننده و یک جمع کننده لازم است. برای انجام عمل ضرب، از الگوریتم ضرب CORDIC استفاده شده است.

برای ساختن یک تابع مولد برای واحد درونیابی خطی، نیاز به محاسبه تمام پارامترهای درونی آن تابع خاص می‌باشد و دقت لازم برای خروجی توسط طول ثباتی که نتیجه در آن ذخیره می‌شود،

- [5] Li Y. and Chu W., "A new non-restoring square root algorithm and its VLSI implementations", IEEE International Conference on Computer Design: VLSI in Computers and Processors (ICCD '96), Pages: 538 – 544, 1996.
- [6] Li Y. and Chu W., "Parallel-array implementations of a non-restoring square root algorithm", IEEE International Conference on Computer Design: VLSI in Computers and Processors (ICCD '97), Pages: 690 – 695, 1997.
- [7] Mayer J. H., "Verification, Synthesis and Implementation of the CORDIC VHDL Code for a FPGA Device", Universitat Autònoma de Barcelona Facultat de Ciències Departament d'Informàtica Unitat de Microelectrònica, University of Ulm Faculty of Engineering Department of Electronics and Microelectronics, Project of 1996.
- [8] Mencer O., S'em'eria L., Morfand M. and Delosme J.M., "Application of Reconfigurable CORDIC Architectures"; Journal of VLSI Signal Processing, Vol. 24, No. 2-3, pp.211-221, March 2000.
- [9] Mencer O., Boullis N, Luk W., Styles H., "Parameterized Function Evaluation for FPGAs", 544-554 Electronic Edition FPL 2001, Belfast, Northern Ireland, UK, August 27-29, 2001; Proceedings, Lecture Notes in Computer Science 2147, ISBN 3-540-42499-7, Springer 2001.
- [10] Montuschi, P.; Mezzalama, P.M., "Survey of square rooting algorithms", IEE Proceedings on Computers and Digital Techniques, Pages: 31 – 40, 1990.
- [11] Parhami B."Computer Arithmetic: Algorithms and Hardware Designs", ISBN 0-19-512583-5, QA76.9.C62P37 ©2000 Oxford University Press, New York
- [12] Tang P.T.P., "Table Lookup Algorithms for Elementary Functions and Their Error Analysis," Proc. 10th IEEE Symposium Computer Arithmetic, IEEE Press, pp. 232–236, 1991.
- [13] Tchoumatchenko V., Vassileva T. and Gurov P. "A FPGA Based Square-Root Coprocessor", Proceedings of the 22nd EUROMICRO 96. Beyond 2000: Hardware and Software Design Strategies, Page(s): 520 -525.
- [14] Tommiska M.T., "Area-efficient implementation of a fast square root algorithm", Third IEEE International Caracas Conference on Devices, Circuits and Systems, Page(s): S18/1 -S18/4, 2000.
- [15] Volder J.E., "The CORDIC Trigonometric Computing Technique," IRE Transactions On Electronic Computers, Vol. EC-8, No. 3, September 1959.
- [16] Rahman A.F.R., Fairhurst M.C., "Enhancing multiple expert decision combination strategies through exploitation of a priori information sources," IEE Proc. Vision Image and Signal Process, 1999.
- [17] Constantinidis A.S., Fairhurst M.C., Rahman A.F.R., "A new multiple-expert decision combination algorithm and its application to the decision of circumscribed," Journal of Pattern Recognition, Elsevier, Page(s):1527-1537, 2001.
- [18] Timarchi S., Miremadi S.G., Ejlali A.R., "Evaluation of Some Exponential Random Number Generators Implemented by FPGA", accepted in the IASTED International Conference on PARALLEL AND DISTRIBUTED COMPUTING AND NETWORKS (PDCN 2005), February 15-17, 2005 Innsbruck, Austria.

$$\frac{23865788.16}{9460566.859} \approx 2.5$$

 جدول (۲): مقادیر $\text{area} \times \text{time}^2$ حاصل از شبیه‌سازی روشها

Method	$\text{area} \times \text{time}^2$	The Rank of Performance
CORDIC	972902919.68	5
PLT	23865788.16	3
LI	447903593.536	4
NRLI	15678991.42	2
NRP	9460566.859	1

۵- نتیجه‌گیری و خلاصه

از آنجا که الگوریتم‌های تکراری، راه حل‌های کندرتی نسبت به الگوریتم‌های تقریبی را پیشنهاد می‌کنند، لذا روش‌های تقریبی بیشتر مورد استفاده قرار می‌گیرند. اما روش‌های تقریبی به دلیل استفاده از LUT برای بدست آوردن یک تقریب اولیه دارای سربار سخت افزاری سیار زیادی هستند. در این مقاله با استفاده از hybrid Multiple expert system، دو روش ارائه گردید که کارایی مدارات محاسبه جذر را بهبود بخشیده شده‌اند. نتایج سنتز مدارات در نوع Xilinx Spartan3 Multiple expert system ترکیبی، روش‌های تولید جذر بهبود قابل ملاحظه‌ای خواهند یافت. همانطور که در جدول ۱ ذکر شده است، روش‌های پیشنهادی با حذف LUT ها و جایگزینی یک اکسپرت جذر به جای این جداول، حجم سخت‌افزاری را سیار بهبود بخشیده‌اند در حالیکه سرعت کاهش یافته است. اما بهبود سربار سخت‌افزاری به اندازه‌ای است که کارایی کل با معیار $\text{area} \times \text{time}^2$ افزایش قابل قبولی دارد. به عنوان مثال مدار پیشنهادی NRLI کارایی روش درونیابی خطی را حدود ۲۸ برابر افزایش داده است. همچنین با استفاده از این روش در ساختار تکه‌ای که منجر به مدار NRP گردید، کارایی روش حدود ۲/۵ برابر افزایش یافت.

مراجع

- [1] Andraka R.; "A Survey of CORDIC Algorithms for FPGAs", Proc ACM/SIGDA International Symposium, Field Programmable Gate Arrays, ACM Press, pp. 191–200, 1998.
- [2] Ibrahim A. A., Elsimary H. A. and Salama A. E.; "FPGA Implementation of Fast Radix 4 Division Algorithm", Proceedings of the 4th IEEE International Workshop on System-on-Chip for Real-Time Applications, 2004.
- [3] Kornerup P., "Digit Selection for SRT Division and Square Root", IEEE TRANSACTIONS ON COMPUTERS, VOL. 54, NO. 3, MARCH 2005, Pages: 294 – 303.
- [4] Lashko A, Zakaznov O; "VHDL implementation of CORDIC algorithm", Master thesis performed in Electronics Systems, Series number: LITH-ISY-EX-3515-2004, 2003-02-20.

زیرنویس‌ها

^۱ Restoring

² Non restoring

³ Piecewise

⁴ Interpolation

⁵ Experts

⁶ Iterative

⁷ Hyperbolic

⁸ Fault tolerant

⁹ Availability

¹⁰ Real time

¹¹ Pipeline