

## یک فرکانس ساز دیجیتال مستقیم بهینه شده (DDFS)

چکیده

طراحی یک فرکانس ساز دیجیتال مستقیم بهینه (DDFS) به لحاظ ROM کا هش یافته، خروجی و سرعت بالا در این مقاله طراحی شده DDFS با فرکانس کلک مرجع 200MHz و FTW 32 بیتی برای تولید سیگنال سینوسی و کسینوسی با فرکانس خروجی 16 بیتی که دارای وضوح فرکانسی  $0.0466\text{Hz}$  و وضوح فازی  $0.0055^\circ$  است، طراحی شده. طراحی DDFS با استفاده از VHDL در Model Sim 10.1d در شبیه سازی شده و بعد با استفاده از ابزار Xilinx ISE 13.2 برای کاربرد در FPGA-Spartan3E ترکیب شده. عملکرد DDFS مبتنی بر ROM کا هش یافته‌ی طراحی شده نشان میدهد که تاخیر CORDIC مبتنی بر ROM معمولی 369ns بهبود یافته. از طریق الگوریتم CORDIC کا هشی با افزایش بیشینه فرکانس تا 138.47 MHz با تاخیر 73ns و کاهش تعداد برشها و LUT آن تا 56% بیشتر بهبود میابد لغات کلیدی : DDFS , ROM , LUT , الگوریتم CORDIC به کارگیری FPGA

### 1 معرفی

تولید کننده‌های سیگنال نقش کلیدی را در تست و اشکال زدایی سیستم‌های الکترونیکی مختلف با تولید شکل موج‌های مختلف که برای تحریک در طول تست لازم است را ایفا میکنند [7]. سیستم‌های ارتباطی بیسیم امروزی مانند تکنیک پخش طیف به سیگنال با سرعت سویچینگ فرکانس بالا و وضوح بهتر در قیاس با سیگنال‌های تولیدی توسط حلقه قفل شده فاز (PLL) های آنالوگ متداول نیاز دارند [5][6]

به علاوه سیگنال‌های درجه 4 برای مدل‌سیون دیجیتال و تبدیل بالا-پایین لازم‌اند. این گونه تولید سیگنال از طریق DDFS قابل انجام است که با نام نوسان گر کنترل شده عددی (NCO) نیز شناخته می‌شوند.

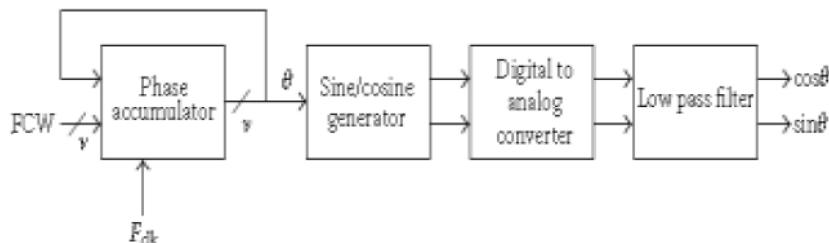
DDFS یک سیگنال سینوسی و کسینوسی را تولید می‌کند و قادر به تغییر فرکانس خروجی با تغییر فرکانس کلک مرجع (Fclk) یا تنظیم واژه تنظیم فرکانس (FTW) که با نام واژه کنترل

فرکانس از آنجا که به عنوان یک سیگنال کنترل کننده عمل میکند نیز شناخته میشود، میباشد [9]

مروری بر DDFS در قسمت بعدی آمده، DDFS مبتنی بر رام در قسمت 3 با روش عادی و کاهشی آمده. قسمت 4 با روش الگوریتم CORDIC سروکار دارد. به کارگیری و نتایج آزمایشی در بخش 5 مطرح و بررسی شده و در بخش 6 جمع بندی میشود

## 2 مروری بر DDFS

بلوک دیاگرام ساده شده DDFS در شکل 1 آمده، DDFS شامل یک جمع کننده فاز (PA) یک تولید کننده سینوسی/کسینوسی، یک مبدل دیجیتال به آنالوگ (DAC) و یک فیلتر است [9]



شکل 1 بلوک دیاگرام ساده شده DDFS

جمع کننده فاز FTW را افزایش میدهد و خروجی آن معرف اطلاعات فاز داده شده به تولید کننده سینوسی/کسینوسی میباشد که تولید مقادیر گستره دامنه‌ی معادل را از مقادیر معادل فاز ورودی انجام میدهد. بعد آن مقادیر با گذر از DAC به آنالوگ تبدیل شده و به فیلتر برای سیگنال های پیوسته سینوسی/کسینوسی به عنوان خروجی تولیدی داده میشوند. فرکانس

$F_{out} = \frac{FTW \cdot F_{clk}}{2^N}$  ----- (1)  
که خروجی DDFS این گونه بdst می آید و ازه تنظیم فرکانس و  $F_{clk}$  فرکانس کلک و  $N$  طول لغت جمع کننده فاز است. فرکانس بیشینه و کمینه تولیدی توسط DDFS از معادله 2 بdst می آید

$$F_{min} = \frac{F_{clk}}{2^N}, \quad F_{max} = \frac{F_{clk}}{2} ----- (2)$$

برای یک فرکانس کلای  $200\text{MHz}$  و جمع کننده فاز 32 بیتی یک فرکانس کمینه  $0.0466\text{Hz}$  که باوضوح فرکانس نیز شناخته میشود و بیشینه فرکانس  $100\text{MHz}$  که محدود به تئوری نمونه برداری نایکویست/شانون است ممکن میباشد

از مقاله [5] ،FTW تعیین کننده سرعت چرخش فاز برای تولید یک دوره کامل موج سینوسی/کسینوسی است.

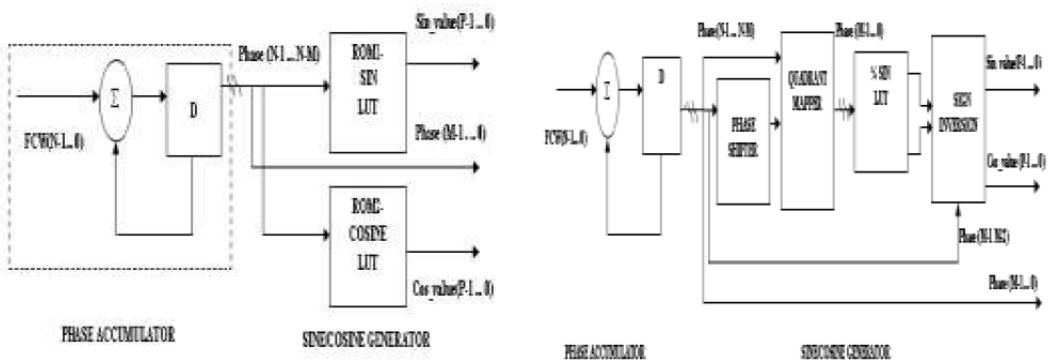
بنابراین سرعت بالاتر تولید موج فرکانس بالا را ممکن میسازد. در DDFS های متداول ، تبدیل فاز به دامنه از طریق جدول جست و جو (LUT) انجام میگیرد. افزایش طول خروجی یا وضوح مقادیر موجود در ROM برای عملکرد بالاتر به اندازه خیلی بزرگی نیاز دارد که قابل قبول نیست چون کاهش سرعت با افزایش محیط و تلفات توان [4] و الزامات طراحی DAC وضوح بالا. بنابراین نیازی برای کاهش اندازه ROM برای بهینه سازی بهتر DDFS وجود دارد

### 3 DDFS های مبتنی بر ROM

در این روش ROM به عنوان LUT شامل مقادیر سینوسی/کسینوسی که توسط اطلاعات فاز برای تبدیل از فاز به مقادیر معادل دامنه استفاده میشود عمل میکند. ROM با  $J^1 X^2$  تعریف میشود که  $J$  و  $X$  طول واژه ورودی و خروجی ROM میباشد. خروجی  $N$  بیت جمع کننده فاز را می توان به  $M$  بیت برای کاهش اندازه ROM کوتاه کرد ولی کوتاهی به خطای کوتاهی فاز منتهی میشود که در قبال مزایای کاهش اندازه ROM قابل قبول است [3][2] ولی  $M$  بیت کاملاً اندازه LUT را کاهش نمی دهد.

#### 3.1 های مبتنی بر ROM معمولی

برای تولید موج سینوسی/کسینوسی همزمان از DDFS نیاز به 2 ROM با اندازه مشابه  $J^1 X^2$  است که موج خروجی در تعداد  $J$  بیت با وضوح  $^2$  خواهد بود. بلوک دیاگرام آن در شکل 2 نشان داده شده. برای مقدار فاز  $M$  بیت کوتاه شده ، مقادیر معادل دامنه سینوسی/کسینوسی  $P$  بیت آن توسط آدرس دهی استاتیک ROM خروجی داده میشوند.



شکل 2 ROM معمولی

شکل 3 ROM کا هشی

### 3.2 DDFS مبتنی بر ROM کا هشی

اندازه ROM را میتوان با استفاده از خواص مثلثاتی کا هش داد [1] با استفاده از خواص مثلثاتی متقارن موج سینوسی که مبتنی بر ROM کا هشی همچنان 2 ROM دارد ROM کسینوسی را میتوان کاملا با استفاده از خاصیت جابه جایی فاز که در آن از  $\sin(\pi i + \theta) = -\sin(\theta)$ ,  $\sin(\pi i - \theta) = \sin(\theta)$  و غیره است اندازه ROM را میتوان از  $J^2 X^{l-2}$  به  $J^2 X^l$  کا هش داد در قیاس با ROM معمولی ، مبتنی بر ROM کا هشی همچنان 2 ROM دارد ROM کسینوسی را حذف کرد. مقادیر کسینوسی را میتوان از  $\sin(\pi/2 + \theta) = \cos(\theta)$  و یافتن مقادیر سینوسی برای زاویه جابه جا شده بدست آورد بلوک دیاگرام کا هشی ROM معمولی در شکل 3 نشان داده شده .

### 4 CORDIC مبتنی بر

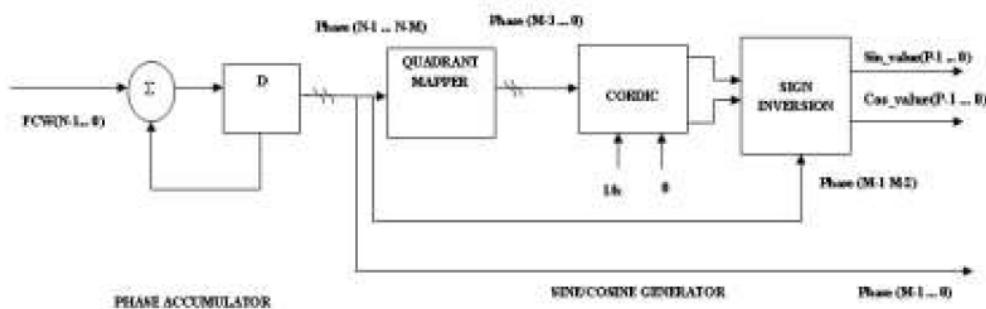
کامپیوتر دیجیتال چرخش مختصات (CORDIC) از تعداد کمتری از سخت افزار ها مانند شیفتراها و جمع کننده ها / کم کننده ها برای محاسبه توابع مثلثاتی استفاده می کند. مختصات برداری را حول دایره واحد از زوایای ثابتی می چرخاند تا زمانی که زاویه آن به صفر کا هش یابد. الگوریتم Volder [10] مقادیر بردار چرخشی را با چرخاندن بردار  $[X_i, Y_j]$  به  $[X_{i+1}, Y_{j+1}]$  از معادلات کلی بدست می آورد

$$x' = x \cos(\theta) - y \sin(\theta); \quad y' = y \cos(\theta) + x \sin(\theta) \quad ---- (3)$$

معادلات خروجی نهیی از تکرار n ام

$$X_n = K [x_i \cos Z_i - y_i \sin Z_i]; Y_n = K [y_i \cos Z_i + x_i \sin Z_i] \quad \text{---- (4)}$$

که بعد از  $n=16$  تکرار  $k=1.607253$  در طول هر تکرار مقایسه ای میان زاویه درخواستی اولیه و زاویه نتیجه شده انجام شده و بعد بر مبنای نتایج مقایسه علامت برای چرخش بعدی تعیین می گردد [8] برای تولید موج سینوسی/کسینوسی با روش مبتنی بر CORDIC، خروجی جمع کننده فاز به عنوان زاویه چرخشی دلخواه در کنار مقادیر  $x$  و  $y$  اولیه به ترتیب به عنوان  $1/k$  و ۰ قرار گرفته و به بلوک CORDIC داده می شوند. از آنجاکه الگوریتم حول  $\pi/2 - \pi/2$  کانورج میکند ۲ MSB بیت جمع کننده فاز برای تولید موج برای دوره ۰ تا  $2\pi$  استفاده شده با نقشه دهی درجه ۴ و معکوس کردن علامت برای مقادیر زاویه ای دلخواه در توابع درجه ۴ طبق شکل ۴ باید انجام شود

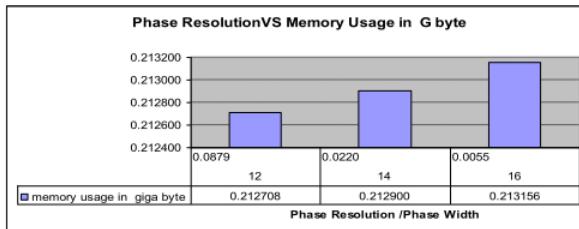


شکل ۴ : بلوک دیاگرام DDFS مبتنی بر CORDIC

### 5 بکارگیری و نتایج آزمایشی

شبیه سازی با FTW 32 بیتی و فرکانس کلاک 200MHz با تغییر طول فاز موج با طول دامنه 12 و 16 بیت انجام میشود برای فرکانس خروجی 10MHz و 5MHz، FTW ی 214748364 و 107374182 باید داده شود که میتوان از محاسبه معادله ۱ بدست آورد

مشاهده میشود که فرکانس موج خروجی از یکی به دیگری بلافاصله با تغییر FTW همانطور که در شکل ۶ نشان داده شده عوض میشود. معماری طراحی شده را میتوان با استفاده از زبان VHDL در FPGA بکار برد و با استفاده از ابزار XILINX ISE 13.2 برای دستگاه هدف Spartan 3E FPGA شبیه سازی کرد

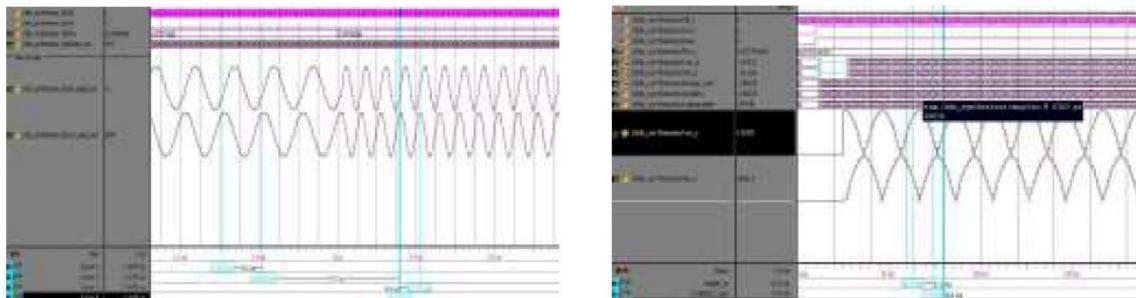


	Maximum Frequency in MHz	Number of slices	Number of 4 Input LUTs
Reduced ROM based DDFS	67.788	799	1516
Pipelined CORDIC based DDFS	138.466	450	848

شکل 5 وضوح در برار استفاده حافظه

جدول 1 عملکرد DDFS

عملکرد DDFS مبتنی بر ROM را می توان از طریق زمان لازم برای خروجی بیشینه بعد از 6.492ns برای ROM معمولی و 6.123ns برای ROM کا هشی بررسی کرد. بنابراین تاخیر به اندازه 369ns بهبود می یابد. هرچه پهنای فاز بیشتر ، مساحت بیشتری را میتوان از طریق افزایش استفاده از حافظه با مزیت افزایش وضوح خروجی از شکل 5 مشاهده نمود.



شکل 6 DDFS مبتنی بر CORDIC کا هشی

برای بهبود بیشتر در عملکرد ، الگوریتم CORDIC لوله ای در VHDL توسعه داده شد. برای زاویه ورودی 16 بیتی  $108^\circ = 19660$  بلوک CORDIC سینوس و کسینوس  $108^\circ$  را با وضوح فرکانس 0.04657Hz و وضوح فاز  $0.0055^\circ$  میدهد از انجاکه ROC  $\pm 90^\circ$  است، زاویه به  $13104(72^\circ)$  نقشه دهی شده و بلوک CORDIC خروجی را برای زاویه سینوسی و کسینوسی به ترتیب 31166 و 10122 میدهد اما کسینوس  $108^\circ$  نیاز به علامت منفی دارد که در مبدل علامت انجام شده و خروجی های آن به ترتیب 31166 و 10122- به عنوان خروجی DDFS میشوند. در قیاس با تکنیک های دیگر، سرعت با فرکانس بیشینه 138.47MHz افزایش میابد و همچنین تعداد برشها و LUT ها همانطور که در جدول 1 نشان داده شد کا هش میابد

## 6 نتیجه گیری

این مقاله نشان میدهد که در قیاس با ROM معمولی DDFS مبتنی بر ROM کاهاشی با بهره گیری از خواص مثلثاتی بهتر است اما الگوریتم CORDIC عملکرد آن را با خروجی بیشتر بهتر می کند ROM کاهاشی طراحی شده تا خیر سیستم ROM کاهاشی را 369ns کاهاش میدهد و DDFS مبتنی بر CORDIC را با کاهاش تعداد برش ها و LUT های DDFS مبتنی بر ROM کاهاشی به اندازه 73ns با بیشینه فرکانس 138.47MHz بیشتر کاهاش میدهد . بنابراین DDFS پیشنهادی به لحاظ سرعت با وجود استفاده از همان جمع کننده فاز برای هر دو روش با اندازه بزرگتر برای وضوح بیشتر. این نتایج برای فرکانس خروجی بیشتر با استفاده الگوریتم مقیاس آزاد لوله ای CORDIC و بکارگیری آن در شبیه سازی ASIC بیشتر بهبود میابند کارهای اتی در ارتباط با شبیه سازی ASIC است که میتوان تخمین توان را انجام داد و بکارگیری DDFS در سیستم های ارتباطی پخش طیف.

## منابع

- [1] Ching-Yuan Yang, “A 5-GHz Direct Digital Frequency Synthesizer Using an Analog-Sine-Mapping Technique in 0.35-um SiGe BICIMOS, IEEE Journal of Solid-States Circuits, 2011.
- [2] Davide De Caro, Digital Synthesizer/Mixer with Hybrid CORDIC–Multiplier Architecture: Error Analysis and Optimization, IEEE transactions on circuits and systems, 2009
- [3] B. G. Goldberg, Direct Digital Frequency Synthesis Demystified.Eagle Rock, VA: LLH Technol. Publ., 1999.

[4] Jouko Vankka, "Digital Synthesizers and Transmitters for Software Radio",  
Editor Springer, 2005.

[5] Jouko Vankka, "Direct Digital Synthesizers: Theory, Design and Applications",  
London, Kluwer Academic Publishers, 2001.

[6] H.T. Nicholas III and H. Samueli, "A 150-MHz direct digital synthesizer in  
1.25 mm CMOS with 90-dBc spurious performances", IEEE Journal of Solid-  
State Circuits, 1991.

[7] D. Peterson and B. Precision, Function Generator and Arbitrary Waveform  
Generator Guidebook. Yorba Linda, CA: B&K Precision Corporation, 2010.

[8] N. Takagi, T. Asada, and S. Yajima, "Redundant CORDIC methods with a  
constant scale factor for sine and cosine computation", IEEE Transactions on  
Computers, 1991.

[9] Tze-Yun Sung, "Low-power and high-SFDR direct digital frequency  
synthesizer based on hybrid CORDIC algorithm", 2009 IEEE International  
Symposium on Circuits and Systems, 2009.

[10] J. E. Volder, "The CORDIC trigonometric computing technique," IRE  
Transactions on Electronic Computers, vol. 8, no. 3, pp. 330–334, 1959.