

This technical report is available in **English** as a PDF file named **Report_English** along with the project files.

نام پروژه : آشکارساز پالس و تخمین فرکانس مبتنی بر FPGA (صفحه پروژه در گیت‌هاب)

نام طراح : میلاد زادرفیع

تاریخ ایجاد و نسخه پروژه : آذر ۱۴۰۴ - ورژن ۱.۰

نرمافزار و زبان طراحی : Xilinx ISE 14.7 – VHDL

راههای ارتباطی با طراح :

<https://www.linkedin.com/in/zadrafee>

<https://www.github.com/zadrafee>

این مستند بخشی از مجموعه پروژه‌های LogLab FPGA است که در وبسایت LogLab منتشر می‌شود. برای مشاهده سایر پروژه‌های مرتبط و مقالات فنی، به وبسایت رسمی من مراجعه کنید:

<http://www.LogLab.ir>

کد توصیف سختافزار آورده شده در این مقاله به همراه تمامی فایل‌ها و پیوست‌ها که توسط نویسنده، میلاد زادرفیع، در گیت‌هاب، لینکدین و وبسایت LogLab منتشر شده است، صرفاً به منظور به اشتراک گذاشتن نمونه‌ای از پروژه‌هایی است که اینجانب انجام داده است و طراح کد هیچ گونه مسئولیت قانونی، مالی، فنی یا عملیاتی ناشی از استفاده، تغییر یا توزیع این کد را نمی‌پذیرد و استفاده از آن با مسئولیت خود کاربر است. **بازنثر و استفاده از محتوای این پروژه تنها با ذکر منبع مجاز است.**

شرح فایل‌ها و فولدرهای اصلی پروژه

Frequency_Estimation_MATLAB	ifm.slx	شبیه‌سازی در Simulink
Frequency_Estimation_VHDL	IF_FFT.xise	فایل پروژه در ISE
	IF_FFT_Top.vhd	مazzoل اصلی
	IQ_Demodulator.vhd	مazzoل دمدولاتور
	Noise_Mean_Calculator.vhd	مazzoل متوضط گیر نویز
	IF_IFF_Top_tb.vhd	مazzoل تست بنج
	Report_Persian.pdf	گزارش فنی به زبان فارسی
	Report_English.pdf	گزارش فنی به زبان انگلیسی

چکیده

هدف از انجام این پروژه، تخمین فرکانس سیگنال IF به کمک تبدیل FFT در حوزه دیجیتال است. سیگنال IF ورودی دارای فرکانس مرکزی 160Mhz و پهنه‌ای باند 40Mhz است. این سیگنال با فرکانس 128Mhz و رزولوشن 14 بیت به صورت مستقیم نمونه برداری می‌شود و برای پردازش‌های بعدی به FPGA فرستاده می‌شود.

فرض بر این است که سیگنال ورودی به صورت پالسی است و عرض پالس‌ها حداقل 200ns است و سیگنال به نویز ورودی حداقل 15dB است. این سیستم باید بتواند پالس‌های دریافتی را آشکار کند و سپس فرکانس آنها را به کمک تبدیل FFT محاسبه کند. در صورتی که سیگنال دریافتی، موج پیوسته باشد باید در بازه‌های زمانی مشخصی مقدار فرکانس را محاسبه و اعلام نماید.

مقدمه

مدولاسیون فرایندی است که در آن ویژگی‌های یک سیگنال حامل، مانند دامنه، فرکانس یا فاز، بر اساس یک سیگنال پیام تغییر داده می‌شوند تا امکان انتقال اطلاعات از طریق یک کانال مخابراتی فراهم شود. این فرایند باعث می‌شود سیگنال پیام که معمولاً فرکانس پایینی دارد، به باند فرکانسی بالاتری منتقل شود تا در برابر نویز مقاوم‌تر شده و قابلیت ارسال در مسافت‌های طولانی یا از طریق محیط‌های مختلف را داشته باشد.

در سیستم‌های مخابراتی، سیگنال RF همان سیگنال حامل مدوله شده است که اطلاعات Baseband روی آن سوار شده است. به طور دقیق‌تر سیگنال RF به سیگنال الکترومغناطیسی گفته می‌شود که در باند امواج رادیویی، معمولاً از چند کیلوهرتز تا چندین گیگاهرتز، قرار دارد. این سیگنال برای انتقال بی‌سیم اطلاعات استفاده می‌شود و به گونه‌ای طراحی می‌شود که بتواند در فضا منتشر شود، توسط آنتن تولید و دریافت گردد، و ویژگی‌های آن مانند توان، پهنه‌ای باند و طیف قابل کنترل و پردازش باشد.

سیگنال IF یا «فرکانس میانی» سیگنالی است که در یک گیرنده یا فرستنده، پس از انجام مدولاسیون یا دمدولاسیون اولیه، به یک فرکانس میانی ثابت تبدیل می‌شود تا پردازش، فیلترگذاری یا تقویت آن ساده‌تر و پایدارتر انجام شود. استفاده از فرکانس میانی باعث بهبود دقت فیلترها، افزایش حساسیت گیرنده و سهولت طراحی سیستم‌های مخابراتی می‌شود.

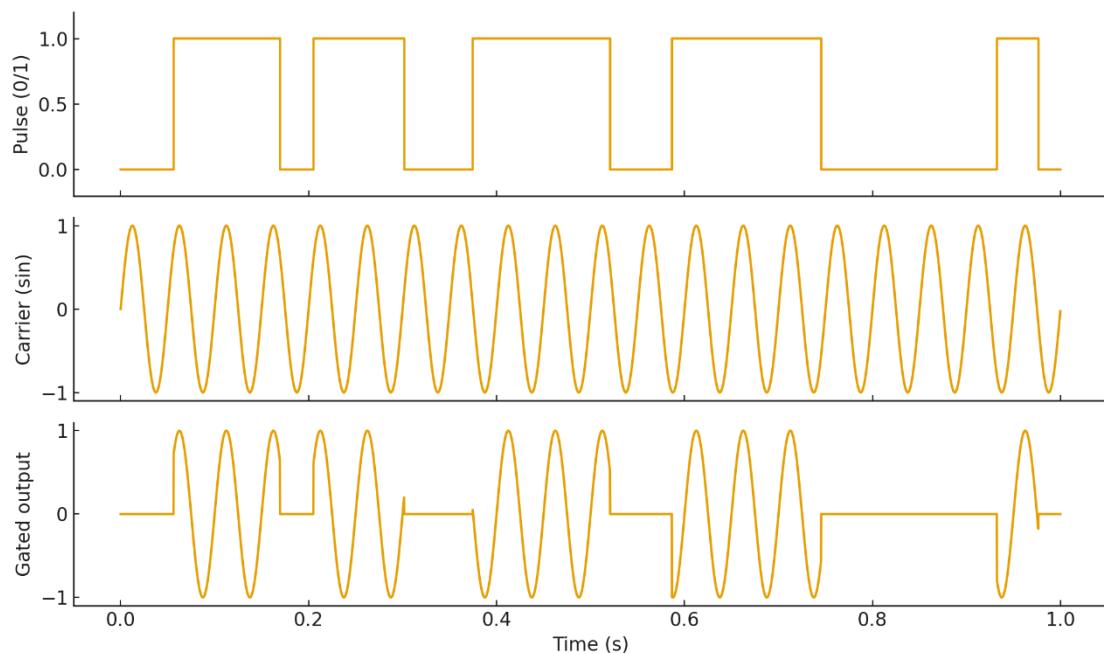
سیگنال Baseband به سیگنال اصلی حامل اطلاعات گفته می‌شود که در کمترین باند فرکانسی ممکن قرار دارد و قبل از هرگونه دمدولاسیون تولید می‌شود. این سیگنال می‌تواند شامل داده‌های دیجیتال، موج صوتی، یا هر سیگنال پیام دیگری باشد. در گیرنده‌ها نیز پس از دمدولاسیون و حذف حامل، سیگنال Baseband بازیابی می‌شود تا مورد تحلیل، آشکارسازی یا پردازش نهایی قرار گیرد.

در سیستم‌های مخابراتی سطوح مختلفی از نمونه‌برداری وجود دارد. به طور مثال در بعضی از سیستم‌ها ابتدا سیگنال RF به طور آنالوگ Down Convert شده و تبدیل به سیگنال میانی IF شده و سپس مجدد یک

مرحله دیگر Down Convert شده و تبدیل به سیگنال Baseband می‌شود و از سیگنال Baseband نمونه‌برداری می‌شود (به کمک مبدل A/D تبدیل به نمونه دیجیتالی می‌شود). اما در بعضی از سیستم‌های مخابراتی همچون این پروژه، سیگنال RF به طور آنالوگ Down Convert شده به سیگنال IF و به طور مستقیم از سیگنال IF نمونه‌برداری می‌شود به چنین سیستم‌های IF Sampling IF Processing یا گفته می‌شود.

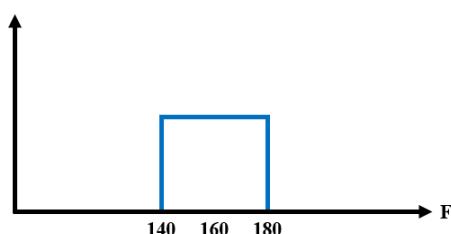
توصیف کلی سیستم

در این پروژه عرض پالس‌های دریافتی نامشخص است تنها اطلاعاتی که در مورد عرض پالس‌ها داریم آن است که بیش از $0.2\mu\text{s}$ است. دقت داشته باشید فاصله بین پالس‌ها نیز نامشخص است.



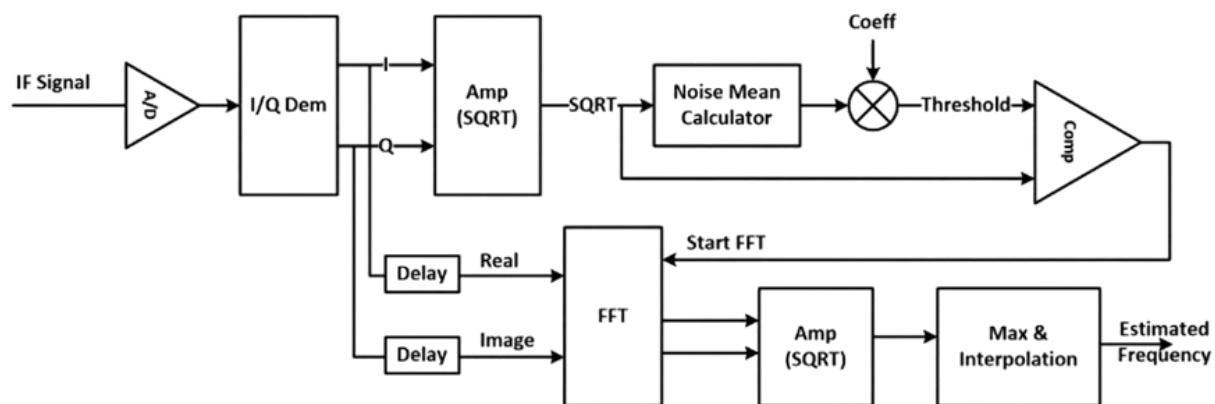
سیگنال IF ای که ما جهت تشخیص پالس دریافت می‌کنیم چیزی شبیه به نمودار سوم (Output) است که نه در سطح RF است و نه در سطح Baseband، در واقع سیگنال پیام ما هنوز سوار بر یک سیگنال حامل است البته این سیگنال حامل در حد RF فرکانس بالایی ندارد بلکه در حد یک سیگنال میانی یا IF است که در این پروژه فرکانس مرکزی IF ما برابر 160Mhz است و همین طور پهنانی باند سیستم ما 40Mhz است.

در صورتی که بخواهیم طیف تغییرات سیگنال حامل بخش IF را مشخص کنیم با توجه به اینکه فرکانس مرکزی IF ما 160Mhz با پهنانی باند 40Mhz است، سیگنال سینوسی که ما دریافت می‌کنیم چیزی در بازه 180Mhz تا 140Mhz است.



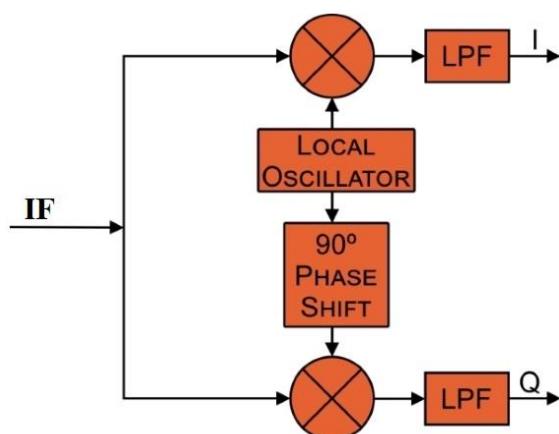
با محاسبات انجام شده در سطح سیستمی، فرکانس نمونهبرداری و همچنین فرکانس کلک مدارمان 128Mhz است. از آنجایی که سیگنال IF ما در بازه 140Mhz تا 180Mhz بوده و فرکانس نمونه برداری ما نیز 128Mhz است با توجه به قضیه نمونهبرداری نایکوییست-شنون نمونهبرداری صحیح زمانی امکان پذیر است که فرکانس نمونهبرداری بزرگتر یا مساوی با دو برابر بزرگترین مولفه فرکانسی سیگنال اصلی باشد با توجه به این که این شرط وجود ندارد در نتیجه امکان نمونهبرداری از این سیگنال نیز وجود ندارد. برای حل این مشکل با روش هایی که در ادامه توضیح داده خواهد شد، شروع بازه فرکانسی سیگنال IF را که 160Mhz است بر روی مقدار 0 تنظیم می کنیم البته پنهانی باند 40Mhz باقی می ماند. با این ترفند بازه فرکانس IF ما به مقدار 0 تا 40Mhz کاهش پیدا می کند، حال این بازه با قضیه نمونهبرداری نایکوییست-شنون سازگار بوده و امکان نمونهبرداری صحیح وجود دارد. در واقع فرکانس نمونهبردای ما بیش از 3 برابر بزرگترین مولفه فرکانسی (40Mhz) است که این مقدار کاملاً کفايت می کند.

بلوک دیاگرام سیستم



قسمت بالایی بلوک دیاگرام که متشکل از بلوک های A/D ، I/Q Demodulator، SQRT، Noise Mean Calculator، Comparator، ضربدر، Coefficient و Threshold است وظیفه آشکارسازی و تشخیص پالس را برعهده دارد و قسمت پایین بلوک بخش تخمین فرکانس است.

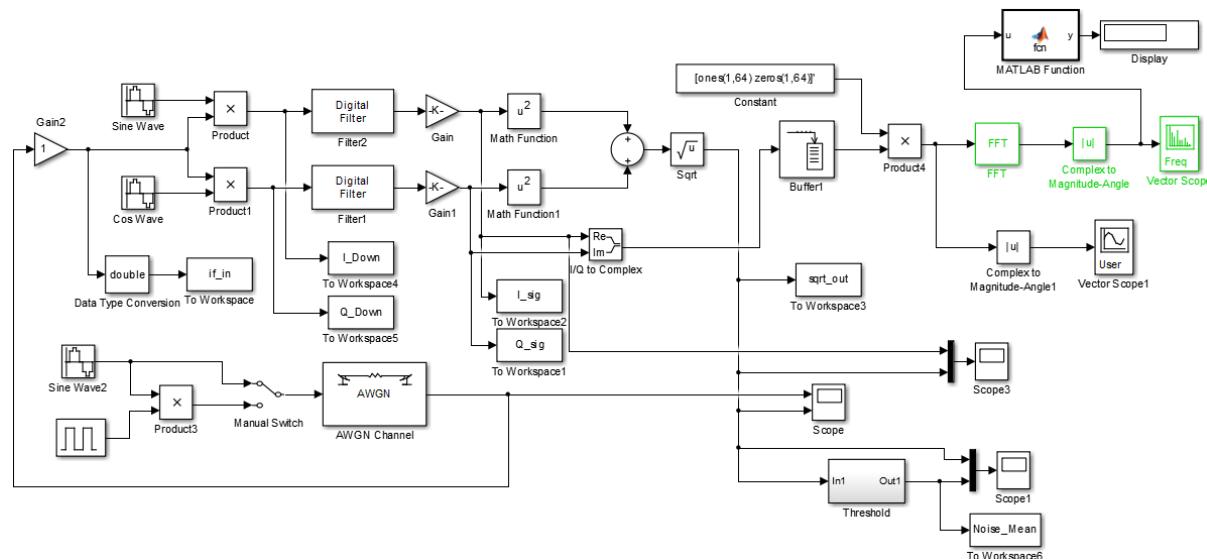
سیگنال آنالوگ IF با عبور از بلوک A/D تبدیل به یک سیگنال گسسته دیجیتال می شود. بلوک I/Q Demodulator سیگنال IF را به سطح Baseband می برد.



در بلوک دیاگرام I/Q Demodulator یک اسیلاتور وجود دارد این اسیلاتور که فرکانس آن 160Mhz است در درون FPGA ساخته می شود و فرکانس با فرکانس سیگنال IF یکسان است. عملکرد این بلوک بدین شکل است که یکبار ورودی را در $\sin 160\text{Mhz}$ ضرب می کنیم و همزمان با 90 درجه شیفت دادن (تاخیر دادن) سیگنال اسیلاتور را تبدیل به $\cos 160\text{Mhz}$ کرده و در ورودی ضرب می کنیم. با این کار طیف سیگنال IF ما شیف پیدا کرده و روی مقدار 0° قرار می گیرد.

طبق تئوری نمونه برداری وقتی از سیگنالی نمونه برداری می کنیم طیف آن سیگنال در حوزه فرکانس در فواصل مشخصی تکرار می شود. از آنجایی که ما از این تکرارها فقط به یک طیف بیشتر نیاز نداریم بنابراین در خروجی ضرب کننده ها از 2 بلوک فیلتر پایین گذر استفاده می کنیم. کار این فیلترهای پایین گذر آن است که تنها یک طیف سیگنال از 40MHz تا 0Hz را برای ما نگه دارد و مابقی تکرارها را فیلتر (حذف) کند.

شبیه‌سازی سیستم در MATLAB نرم‌افزار Simulink



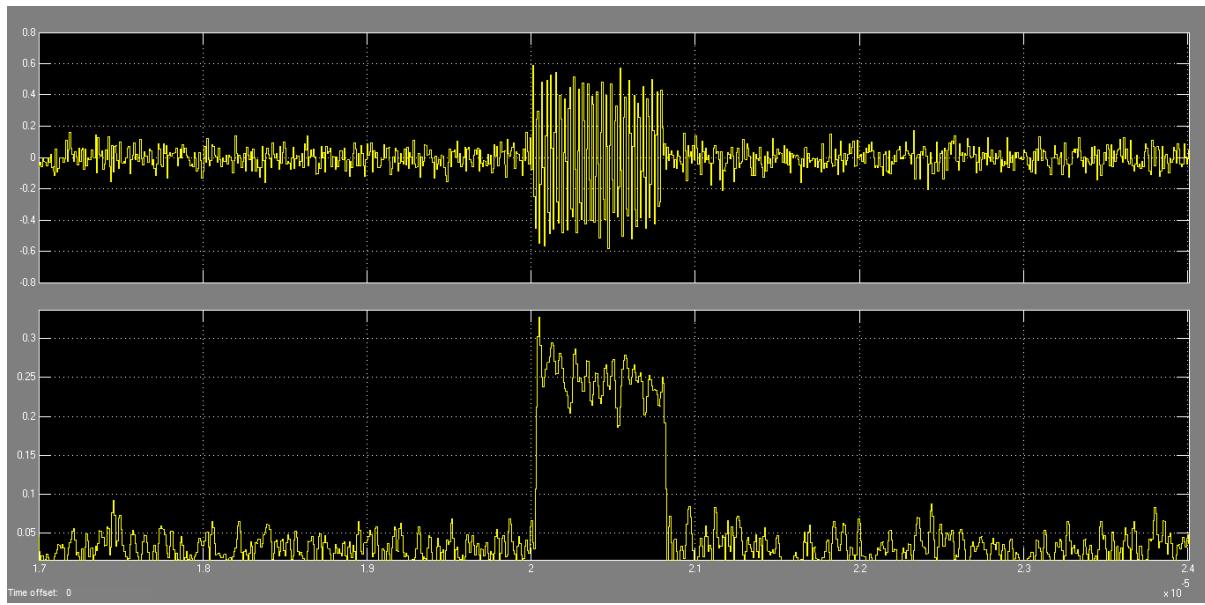
بلوک Amp (SQRT) دامنه سیگنال را از طریق رابطه زیر برای ما محاسبه می کند.

$$\text{Amplitude} = \sqrt{I^2 + Q^2}$$

به عبارت دیگر خروجی بلوک SQRT دیتای Baseband یا پوش سیگنال IF است که برای ما استخراج می‌کند.

در شکل زیر سیگنال IF را به همراه خروجی بلوک I/Q Demodulator مشاهده می کنید:

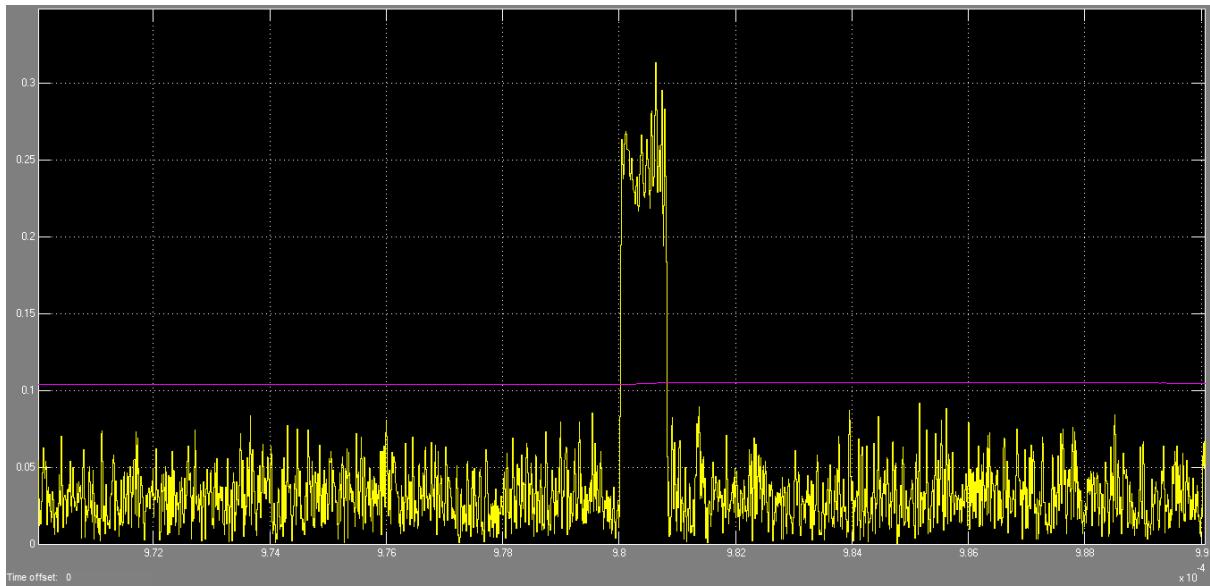
شکل موج بالا تصویر سیگنال IF بوده و شکل موج پایین تصویر سیگنال Baseband



برای اینکه شبیه‌سازیمان به واقعیت نزدیکتر باشد در Simulink بعد از ورودی اصلی یا همان سیگنال پیام از بلوک AWGN استفاده کرده‌ایم با کمک این بلوک به سیگنال‌مان مقداری نویز اضافه می کنیم. (بلوک AWGN مخفف Additive White Gaussian Noise به معنای یک نویز سفید گوسی است که معادل نویزی است که در کانال به سیگنال اضافه می‌شود).

تا اینجا پوش سیگنال را استخراج کردیم حال می‌خواهیم از روی پوش سیگنال پالس‌ها را شناسایی کنیم، برای این منظور از یک مقایسه کننده استفاده می‌کنیم، اما سوالی که در اینجا مطرح می‌شود این است، سیگنال Baseband را باید با چه مقداری مقایسه کنیم؟ با دقت به شکل موج سیگنال Baseband این ایده به ذهنمان می‌رسد که با محاسبه متوسط نویز و چند برابر کردن این مقدار می‌توانیم مرجعی به نام سیگنال Threshold یا آستانه جهت مقایسه با سیگنال Baseband تعریف کنیم که اگر مقدار سیگنال Baseband از مقدار Threshold بیشتر بود به معنای ورود پالس خواهد بود.

اما چگونه متوسط نویز را محاسبه کنیم؟ با کمک یک فیلتر پایین‌گذر، در واقع یک فیلتر پایین‌گذر در عمل می‌تواند به عنوان یک انتگرال‌گیر یا متوسط‌گیر در بحث پردازش سیگنال استفاده شود. پایه بلوک Noise فیلتر پایین‌گذرنی است که در اینجا به آن اشاره شد. Mean Calculator



در تصویر فوق شکل موج زرد رنگ سیگنال Baseband را نشان می‌دهد و شکل موج بنفش رنگ نشان دهنده سیگنال Threshold است. همان طور که مشاهده می‌کنید موقعیت قرارگیری سیگنال Threshold از نویز بالاتر و از پالس پایین‌تر است.

بخش تشخیص فرکانس

تا بدینجا با بخش آشکار سازی پالس را مورد بررسی قرار دادیم از این قسمت به بعد به نحوه تشخیص فرکانس پالس می‌پردازیم. دقت داشته باشید پیش فرض ما در بخش تشخیص فرکانس این است که پالس ما تنها دارای یک فرکانس باشد. پایه اصلی تشخیص فرکانس در این سیستم بلوک FFT است.

بلوک FFT قدرت مولفه‌های فرکانسی موجود در سیگنال ورودی را به ما نشان می‌دهد. همان‌طور که می‌دانیم پالس تشخیص داده شده، بر روی یک سیگنال IF سوار است که در زمان تشخیص دارای فرکانسی بین 140Mhz تا 180Mhz است. با FFT گرفتن از سیگنال پالس باید در آن فرکانس خاصی که سیگنال Baseband روی آن سوار است دامنه بزرگی ببینیم و در سایر نقاط دامنه نزدیک به صفر را مشاهده کنیم در واقع پیک حاصل از خروجی FFT نشان دهنده فرکانس سیگنال ورودی یا همان پالس است.

برای بدست آوردن پیک سیگنال ورودی می‌بایست ابتدا دامنه خروجی‌های FFT را با کمک فرمول زیر محاسبه کنیم. (R و I به ترتیب بخش حقیقی و موهومی خروجی FFT هستند).

$$\text{Amplitude} = \sqrt{R^2 + I^2}$$

با توجه به اینکه فرکانس نمونه برداری ما 128Mhz است و از 128 نقطه استفاده می‌کنیم در نتیجه رزولوشن و دقت نمونه برداری ما برابر 1Mhz می‌شود.

$$\text{Resolution} = \frac{FS}{P} = \frac{128\text{Mhz}}{128} = 1\text{Mhz}$$

این که رزولوشن ما 1Mhz است بدین معناست که اگر فرکانس پالس ما فرضا برابر 10.2Mhz باشد، با ماکزیمم‌گیری یا مقدار 10Mhz را به ما ارائه می‌دهد یا 11Mhz ، برای حل این مشکل و بدست آوردن مقدار دقیق فرکانس پالس می‌توانیم از یک روش محاسباتی به نام پارابولیک اینترپولیشن استفاده کنیم، در پردازش سیگنال، پارابولیک اینترپولیشن یک تکنیک استاندارد برای تخمین دقیق موقعیت پیک در دامنه فرکانس است. در این روش از سه مقدار بیشینه و دو همسایه‌اش در FFT یک سهمی ساخته می‌شود و رأس آن به عنوان فرکانس واقعی پیک تخمین زده می‌شود.

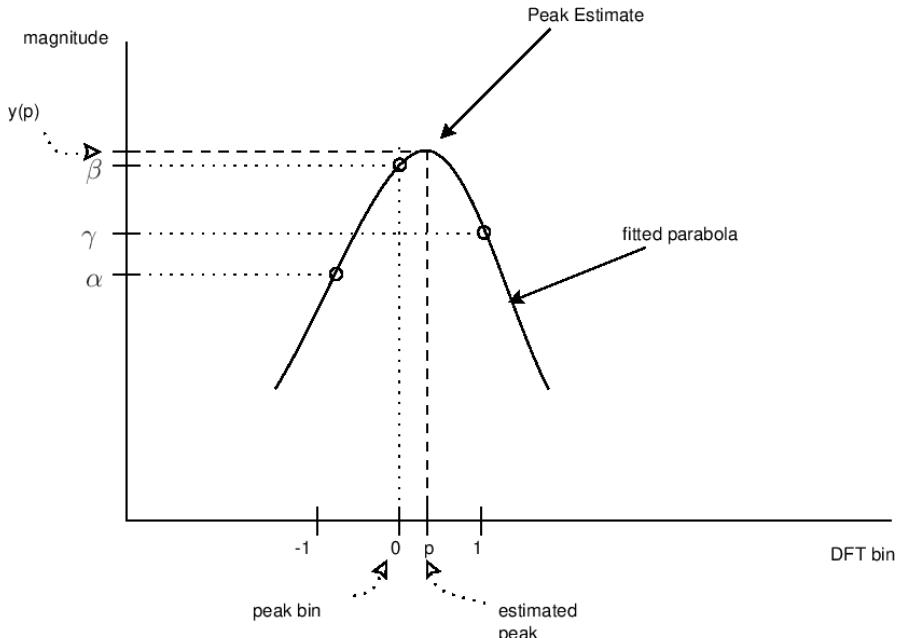


Illustration of parabolic peak interpolation using the three samples nearest the peak.

در حوزه پردازش سیگنال به روش فوق Quadratic Interpolation of Spectral Peaks می‌گویند.

از طریق فرمول زیر مقدار دقیق دامنه را تخمین می‌زنیم.

$$P = \frac{a - c}{2 * (a - 2 * b + c)}$$

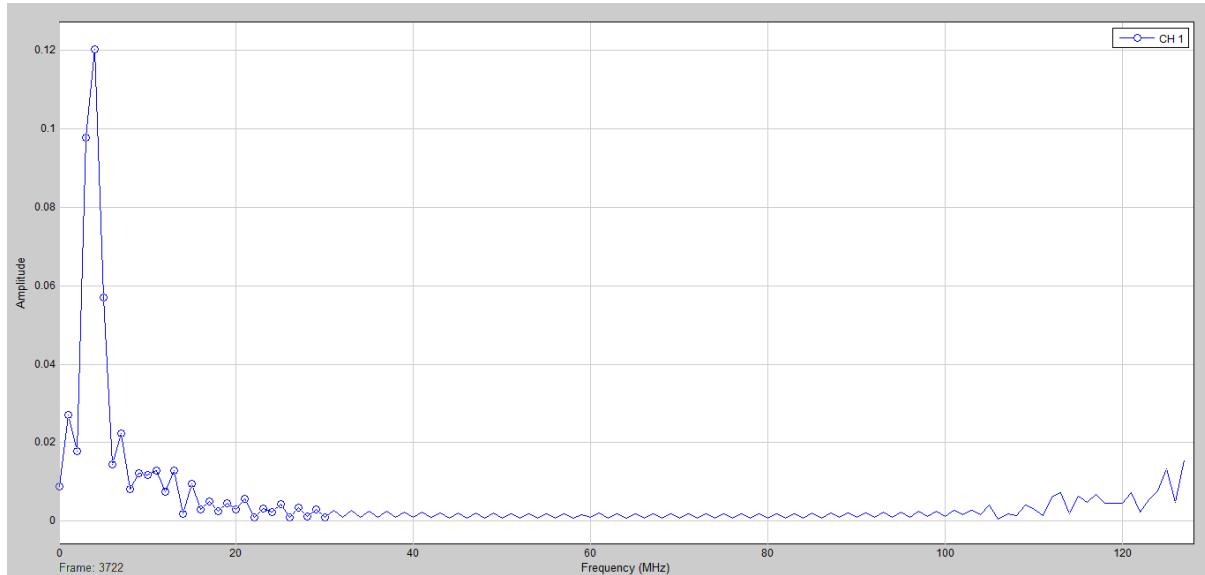
در این فرمول، مقدار b برابر با ماکزیمم دامنه خروجی FFT است و مقدار a برابر دامنه سمت چپ b و مقدار c برابر دامنه سمت راست b است. مقدار فرکانس درون‌یابی شده برابر است با:

$$F_{Estimated} = (k + P) * \frac{f_s}{N}$$

در فرمول فوق k ایندکس متناظر با ماکزیمم دامنه یا به عبارتی ایندکس b است.

جهت آشنایی بیشتر با این روش می‌توانید به [این صفحه از سایت دانشگاه استنفورد](#) مراجعه کنید.

تصویر زیر دامنه خروجی FFT است:



بررسی کد VHDL

در این پروژه ورودی FPGA یک سیگنال 14 بیتی است که این 14 بیت خروجی حاصل از A/D است (دیجیتال شده سیگنال IF) و دارای مازول اصلی IF_FFT_Top، مازول IQ_Demodulator، مازول Low_Pass_Filter و مازول Noise_Mean_Calculator است.

مازول اصلی IF_FFT_Top

این مازول هسته مرکزی سیستم بوده و وظیفه هماهنگ‌سازی مراحل آشکارسازی پالس، استخراج دامنه، اجرای FFT و تخمین دقیق فرکانس را بر عهده دارد. مهم‌ترین بخش‌های عملکرد آن عبارت‌اند از:

دمدولاسیون و استخراج سیگنال Baseband

در این بخش سیگنال IF ورودی ابتدا از طریق مازول IQ_Demodulator به مولفه‌های I و Q تبدیل می‌شود. سپس با استفاده از IP SQRT Cordic که در واقع همان IP است دامنه لحظه‌ای آن توسط رابطه $\sqrt{I^2 + Q^2}$ محاسبه می‌شود و بدین ترتیب سیگنال Baseband ساخته می‌شود.

محاسبه میانگین نویز و تولید آستانه تشخیص پالس

خروجی دامنه Baseband به مازول Noise_Mean_Calculator وارد شده و میانگین نویز استخراج می‌شود. سپس مقدار Threshold چند برابر میانگین نویز تنظیم شده تا لحظه ورود پالس شناسایی شود.

تشخیص شروع و پایان پالس

برای مقایسه دامنه Baseband و Threshold، دو شمارنده مستقل برای شناسایی شروع پالس و پایان پالس استفاده شده‌اند. تنها پس از پایدار بودن سیگنال بالاتر از Threshold، پالس معتبر تشخیص داده شده و FFT_Start فعال می‌شود.

بافر تاخیر برای هم‌تراز کردن I/Q

از آنجایی که فرآیند تشخیص پالس ناگزیر تأخیری ایجاد می‌کند، دو آرایه ۳۱ مرحله‌ای برای ایجاد Delay بر روی سیگنال I و Q استفاده شده تا لحظه شروع FFT دقیقاً با نمونه‌های درست همزمان باشد.

محاسبه دامنه خروجی FFT

پس از اجرای ۱۲۸ FFT نقطه‌ای، بخش حقیقی و موهومی خروجی از طریق IP FFT_SQRT به دامنه تبدیل می‌شود. سیگنال FFT_Data_Valid نقش نمونه‌برداری ریتمیک از خروجی FFT را ایفا می‌کند.

با مشخصات طول ۱۲۸ نقطه، معماری Pipelined و با ورودی‌هایی با عرض بیت ۱۴ بیت و Phase Factor با عرض بیت ۱۶ بیت، به صورت Unscaled پیاده‌سازی شده است.

جستجوی پیک طیف

در هنگام آماده بودن (FFT_Amp_Ready)، دامنه‌ها بررسی می‌شوند تا سه مقدار A، B، C به ترتیب چپ، پیک و راست ذخیره شوند. این مقادیر ورودی الگوریتم Quadratic Interpolation هستند.

$$P = \frac{a - c}{2 * (a - 2 * b + c)}$$

خروجی برای ILA

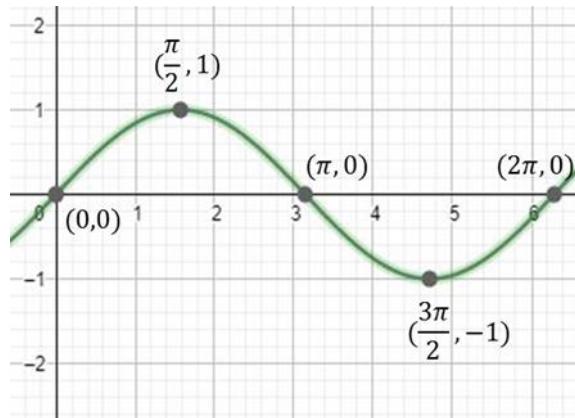
در نهایت ایندکس پیک و Bin Offset در کنار هم جهت مشاهده در ILA قرار می‌گیرند تا روند تخمین فرکانس قابل دیباگ باشد.

نکته در مورد مازول IQ_Demodulator: اگر بخواهیم با فرکانس نمونه برداری ۱۲۸Mhz یک سیگنال سینوسی با فرکانس ۳۲Mhz درست کنیم تنها نیاز به ۴ نقطه داریم.

$$\frac{128\text{Mhz}}{32\text{Mhz}} = 4$$

حسن این روش آن است که دیگر نیاز نیست برای ساخت سیگنال سینوسی از IP DDS استفاده کنیم و مهمتر از آن حتی نیاز به ضرب کننده هم نداریم چرا که سیگنال سینوسی تولید شده تنها مقدار ۰ و ۱ و -۱ را دارد. به طور واضح‌تر جاهايی که سینوس مقدارش '۰' است سیگنال ورودی نیز '۰' می‌شود و

قسمت‌هایی که '1' است خود سیگنال ورودی می‌شود و جاهایی هم که '0' است عکس سیگنال ورودی می‌شود.



فیلترهای پایین‌گذر در IQ_Demodulator

پس از ضرب، مولفه‌های I و Q خام دارای مؤلفه‌های فرکانسی اضافی در $160\text{MHz} \pm \text{Baseband}$ هستند. بنابراین هر مسیر دارای یک فیلتر 5 Tap IIR/FIR ترکیبی است که تنها باند 0 تا 40MHz را حفظ می‌کند.

یادآوری: تپ (Tap) به هر واحد تشکیل‌دهنده فیلتر دیجیتال (در اینجا FIR) گفته می‌شود که شامل یک نمونه تأخیری و یک ضریب (Coefficient) متناظر است. یک فیلتر 5 تپ یعنی فیلتر دارای پنج ضریب و پنج مرحله تأخیر.

Noise_Mean_Calculator

پایه این مازول یک فیلتر پایین‌گذر با پاسخ بسیار آهسته است که برای تخمین میانگین نویز مورد استفاده قرار می‌گیرد.

این بلوک یک Accumulator بزرگ با قابلیت نگهداری «باقیمانده» (Remainder Feedback) است که میانگین متحرک مفید با طول N^2 تولید می‌کند. طول این فیلتر در پروژه برابر Averaging_Length = 15 است، یعنی 32768 نمونه.

برای جلوگیری از اثرگذاری پالس واقعی روی مقدار میانگین نویز، سیگنال Input_Signal_Int_Saturate مقدار ورودی را بررسی کرده و در صورت بزرگ‌تر بودن از مقدار نویز $3 \times$ آن را اشباع می‌کند.

وجود بیت‌های اضافه Extra_Bits = 8 امکان حفظ دقت در محاسبه میانگین را فراهم می‌کند و از بروز خطاهای کوانتش (Quantization) جلوگیری می‌کند.

نکته: روش اشاره شده در بالا جهت محاسبه میانگین نویز و درست کردن مرجعی به نام سیگنال Threshold جهت مقایسه یک روش کاملا علمی و استاندارد است. ولی در پروژه‌های واقعی کمتر از این روش استفاده می‌شود. در عمل با اندازه‌گیری نویز با کمک تجهیزات پیشرفته و بسیار دقیق مقدار واقعی متوسط نویز را اندازه‌گیری نموده سپس با توجه به رنج نویز آن را چند برابر می‌کنند (در این پروژه ۳ برابر) و آن را به عنوان سیگنال Threshold در نظر می‌گیریم. چرا روش تئوری کتاب‌ها را در عمل پیاده‌سازی نمی‌کنیم؟

زیرا در سیستم‌های واقعی نویز دائما در حال تغییر است:

- تغییر با دما
- تغییر با بار ADC
- تغییر با تغییر سطح سیگنال RF
- نویز محیط، EMI، دیجیتال سوئیچینگ
- نویز مبتنی بر کanal، نه نویز سفید

در واقعیت:

- نویز صفر میانگین نیست
- DC offset داریم.
- Leakage از LO داریم.
- Quantization Noise ADC، Harmonic، Spur داریم.
- Drift در LNA/IF Gain داریم.
- AC coupling سر خود Offset ایجاد می‌کند.

در نتیجه «میانگین نویز» اصلاً تعریف دقیق ندارد و منجر به اندازه‌گیری اشتباه می‌شود.

اگرچه در تئوری تعیین آستانه بر اساس میانگین نویز روشی علمی و استاندارد است، اما در عمل به دلیل غیرایستا بودن نویز، وجود آفست‌ها و اسپوریوس‌های سخت‌افزاری، تغییرات محیطی و عدم پایداری نویز واقعی، معمولاً از مقدار Threshold ثابت استفاده می‌شود. مقدار آستانه در ابتدا با اندازه‌گیری تجربی تعیین شده و سپس در سیستم ثابت نگه داشته می‌شود تا عملکرد پایدار و قابل اعتماد تضمین گردد.

فیلتر پایین‌گذر (Low_Pass_Filter) – مسیر I و Q

این فیلتر یک ساختار IIR/FIR ترکیبی (Hybrid Filter) است که هم سریع و هم پایدار است.

ساختار فیلتر

بخش FIR شامل 5 ضرب کننده با ضرایب متقارن $b[n]$ ، بخش IIR شامل فیدبک های با ضرایب $a[n]$ ، ضرایب به صورت 18 بیتی S.2.15 پیاده سازی شده اند که برای پردازش اعداد اعشاری مناسب اند.

دقت بالا و جلوگیری از Overflow

همه ضرب ها با پهنه ای 41 بیت انجام شده و سپس در خروجی Saturation اعمال می شود تا از تجاوز از محدوده عددی جلوگیری شود. خروجی نهایی 14 بیتی است.

وظیفه این فیلتر حذف مولفه های:

160MHz + Baseband

160MHz – Baseband

و سایر هارمونیک ها و تنها عبور دادن محدوده 0 تا 40MHz است.

ماژول Bin_Offset_Calculator

این ماژول یک Divider سخت افزاری است که بخش «Offset» مربوط به Quadratic Interpolation را محاسبه می کند.

ورودی ها:

$$\text{Dividend} = (a - c)$$

$$\text{Divisor} = (a - 2b + c)$$

خروجی:

$$P = \text{Fractional FFT} \quad (\text{آفست بین دو بین})$$

این مقدار به همراه FFT_Max_Index مقدار نهایی فرکانس را تشکیل می دهد.