





دانشکده مهندسی برق و رباتیک
رشته الکترونیک گرایش دیجیتال

پایان نامه کارشناسی ارشد

پیاده‌سازی تبدیل موجک گسسته دوبعدی
روی FPGA

نگارنده: مهدی خدادادی

استاد راهنما
دکتر علی سلیمانی

استاد مشاور
دکتر هادی گرایلو

شهریور ۱۳۹۵



فرم شماره ۷: صورتجلسه دفاع از پایان نامه تحصیلی دوره کارشناسی ارشد

با تأییدات خداوند متعال و با استعانت از حضرت ولی عصر (عج) ارزیابی جلسه دفاع از پایان نامه کارشناسی ارشد خانم / آقای مهدی خدادادی به شماره دانشجویی ۹۲۰۵۷۸۴ رشته مهندسی برق گرایش الکترونیک (دیجیتال) که در تاریخ ۹۵/۰۶/۱۵ تحت عنوان:

پیاده سازی تبدیل موجک گسسته دوبعدی روی FPGA

با حضور هیأت محترم داوران در دانشگاه صنعتی شاهرود برگزار گردید به شرح ذیل اعلام می‌گردد:

قبول (با درجه: خوب - امتیاز ۱۶/۵۷۶) <input checked="" type="checkbox"/>	دفاع مجدد <input type="checkbox"/>	مردود <input type="checkbox"/>
نوع تحقیق: نظری <input checked="" type="checkbox"/>	عملی <input type="checkbox"/>	

- ۱- عالی (۲۰ - ۱۹)
- ۲- بسیار خوب (۱۸/۹۹ - ۱۸)
- ۳- خوب (۱۶ - ۱۷/۹۹)
- ۴- قابل قبول (۱۴ - ۱۵/۹۹)
- ۵- نمره کمتر از ۱۴ غیر قابل قبول

اعضاء	مرتبه علمی	نام و نام خانوادگی	عضو هیأت داوران
	استادیار	سید علی حسینی	۱- استاد راهنمای اول
-	-	-	۲- استاد راهنمای دوم
	استادیار	هادی کریمی	۳- استاد مشاور
	استادیار	علی قانع	۴- نماینده شورای تحصیلات تکمیلی
	استادیار	ایرینا سرزقی	۵- استاد ممتحن اول
	استادیار	محمد رضا آروف	۶- استاد ممتحن دوم

نام و نام خانوادگی رئیس دانشکده:

تاریخ و امضاء و مهر دانشکده:

تشکر و قدردانی

از خداوند مهربان تشکر می‌کنم، که با تمام نافرمانی‌هایم، در همه‌ی لحظات زندگی یار و یاورم بوده است.

از پدر و مادر عزیزم تشکر می‌کنم، که همیشه پشتیبان و مشوقم بوده‌اند.

از استاد راهنما و مشاورم تشکر می‌کنم که با تمام درگیری‌های خود، بی‌دریغ یار و یاورم بودند.

از تمام دوستانم تشکر می‌کنم، که در تهیه‌ی پایان‌نامه کمکم کرده‌اند.

تعهد نامه

اینجانب مهدی خدادادی..... دانشجوی دوره کارشناسی ارشد رشته

.....الکترونیک..دیجیتال..... دانشکده برق.....

دانشگاه صنعتی شاهرود نویسنده پایان نامه پیاده سازی تبدیل موجک گسسته دو بعدی

روی.....FPGA..... تحت راهنمایی..... دکتر علی سلیمانی..... متعهد می شوم .

- تحقیقات در این پایان نامه توسط اینجانب انجام شده است و از صحت و اصالت برخوردار است .
- در استفاده از نتایج پژوهشهای محققان دیگر به مرجع مورد استفاده استناد شده است .
- مطالب مندرج در پایان نامه تاکنون توسط خود یا فرد دیگری برای دریافت هیچ نوع مدرک یا امتیازی در هیچ جا ارائه نشده است .
- کلیه حقوق معنوی این اثر متعلق به دانشگاه صنعتی شاهرود می باشد و مقالات مستخرج با نام « دانشگاه صنعتی شاهرود » و یا « Shahrood University of Technology » به چاپ خواهد رسید .
- حقوق معنوی تمام افرادی که در به دست آمدن نتایج اصلی پایان نامه تأثیرگذار بوده اند در مقالات مستخرج از پایان نامه رعایت می گردد.
- در کلیه مراحل انجام این پایان نامه ، در مواردی که از موجود زنده (یا بافتهای آنها) استفاده شده است ضوابط و اصول اخلاقی رعایت شده است .
- در کلیه مراحل انجام این پایان نامه، در مواردی که به حوزه اطلاعات شخصی افراد دسترسی یافته یا استفاده شده است اصل رازداری ضوابط و اصول اخلاق انسانی رعایت شده است .

تاریخ

امضای دانشجو

مالکیت نتایج و حق نشر

- کلیه حقوق معنوی این اثر و محصولات آن (مقالات مستخرج ، کتاب ، برنامه های رایانه ای ، نرم افزار ها و تجهیزات ساخته شده است) متعلق به دانشگاه صنعتی شاهرود می باشد . این مطلب باید به نحو مقتضی در تولیدات علمی مربوطه ذکر شود .
- استفاده از اطلاعات و نتایج موجود در پایان نامه بدون ذکر مرجع مجاز نمی باشد.

چکیده

تبدیل موجک ابزاری کارآمد جهت کاربردهای مختلف در پردازش سیگنال است و در بسیاری از زمینه‌ها مانند پردازش صدا و ویدئو، حذف نویز تصاویر، شناسایی الگو، کدگذاری و فشرده‌سازی تصویر کاربرد دارد. به همین دلیل ارائه راهکارهایی جهت افزایش سرعت پیاده‌سازی تبدیل موجک مورد توجه قرار گرفته است که یکی از بهترین راهکارها استفاده از پردازش موازی و تراشه‌های FPGA است. در این پایان‌نامه یک معماری جدید برای پیاده‌سازی تبدیل موجک گسسته دوبعدی جهت کاربرد در فشرده‌سازی تصویر ارائه شده است. در این پژوهش از روش طرح بالابر که روش جدیدتر و کاراتری نسبت به روش بانک‌های فیلتری است و همچنین از فیلتر موجک ۵/۳ که در فشرده‌سازی تصویر بدون اتلاف کاربرد دارد استفاده شده است. معماری جدید با هدف کاهش پیچیدگی سخت‌افزاری و بهینه‌سازی تعداد قطعات مصرفی و افزایش فرکانس کاری پیاده‌سازی شده است و شامل یک واحد پردازنده جهت محاسبه ضرایب تبدیل موجک گسسته دوبعدی و یک واحد کنترل جهت کنترل جریان داده در پردازنده و تولید خطوط آدرس حافظه و یک واحد حافظه خارجی جهت ذخیره‌سازی ضرایب تبدیل موجک است. این معماری با زبان برنامه‌نویسی سخت‌افزاری VHDL و با استفاده از نرم‌افزار Xilinx ISE که یک نرم‌افزار قدرتمند جهت برنامه‌نویسی FPGA است نوشته و شبیه‌سازی شده است و نتایج بدست آمده مشاهده و مورد ارزیابی قرار گرفته و با سایر معماری‌ها مقایسه شده است.

برنامه تبدیل موجک گسسته دوبعدی با روش طرح بالابر با زبان متلب نیز پیاده‌سازی و شبیه‌سازی شده است و تصاویر خروجی در هر دو محیط متلب و FPGA با یکدیگر مقایسه شده است.

واژگان کلیدی: تبدیل موجک گسسته- بانک‌های فیلتری- طرح بالابر- کدگذاری زیرباندی-FPGA

- Xilinx ISE - موجک ۵/۳

فهرست مطالب

فصل اول: پیشگفتار.....	۱
۱-۱ تعریف مساله و هدف از تحقیق	۲
۲-۱ مروری بر فصل‌های پایان‌نامه	۳
فصل دوم: مروری بر پژوهش‌های گذشته.....	۵
۱-۲ مقدمه	۶
۲-۲ پژوهش‌های گذشته	۶
فصل سوم: مبانی تئوری تبدیل موجک.....	۱۹
۱-۳ مقدمه	۲۰
۲-۳ چرا تبدیل موجک؟	۲۰
۳-۳ کدگذاری زیر بانندی	۲۱
۴-۳ تبدیل موجک	۲۳
۱-۴-۳ تبدیل موجک گسسته.....	۲۵
۲-۴-۳ نمایش چند درجه تفکیک.....	۲۶
۳-۴-۳ تبدیل موجک و بانک فیلتری.....	۳۱
۴-۴-۳ اعمال تبدیل موجک بر روی تصویر [۱۶].....	۳۳
۵-۴-۳ تبدیل موجک به روش طرح بالابر.....	۳۵
فصل چهارم: روش پیشنهادی برای پیاده‌سازی تبدیل موجک گسسته دوبعدی.....	۳۷
۱-۴ مقدمه	۳۸
۲-۴ نرم‌افزار Xilinx ISE	۳۸
۳-۴ معماری پیشنهادی.....	۳۸
۱-۳-۴ معماری ارائه شده.....	۳۹
۲-۳-۴ واحد پردازنده.....	۴۰
۳-۳-۴ واحد کنترل.....	۴۲
۴-۳-۴ واحد حافظه.....	۴۵
فصل پنجم : نتایج حاصل از شبیه‌سازی و مقایسه با کارهای انجام شده.....	۴۷

۴۸.....	۱-۵ مقدمه
۴۸.....	۲-۵ نتایج
۵۰.....	۱-۲-۵ تصاویر خروجی
۵۷.....	۲-۲-۵ فرکانس کاری و منابع استفاده شده
۵۸.....	۳-۵ مقایسه نتایج برنامه با سایر مقالات
۶۱.....	فصل ششم: نتیجه‌گیری و پیشنهادها
۶۲.....	۱-۶ مروری بر پژوهش انجام شده
۶۲.....	۲-۶ پیشنهادها
۶۳.....	پیوست‌ها
۶۹.....	منابع

فهرست شکل ها

- شکل ۱-۲: مدل ارائه شده برای پیاده سازی تبدیل موجک در [۱] ۶
- شکل ۲-۲: نمایش بلوکی یک مرحله از طرح بالابر با استفاده از معماری ارائه شده در [۲] ۸
- شکل ۳-۲: معماری ارائه شده توسط مرجع [۳] ۹
- شکل ۴-۲: مدل ارائه شده توسط مرجع [۵] ۱۰
- شکل ۵-۲: مدل پیشنهادی تبدیل موجک دوبعدی مرجع [۶] ۱۱
- شکل ۶-۲: مدل تبدیل موجک یک بعدی پیشنهادی [۷] ۱۲
- شکل ۷-۲: نمایش بلوکی مدل پیشنهادی مرجع [۸] ۱۳
- شکل ۸-۲: نمایش بلوکی معماری ارائه شده در [۹] ۱۴
- شکل ۹-۲: معماری پیشنهادی در مرجع [۱۰] ۱۶
- شکل ۱۰-۲: مدار مربوط به پردازنده در معماری پیشنهادی مرجع [۱۰] ۱۶
- شکل ۱-۳: یک نمونه بانکهای فیلتری میان گذر ۲۳
- شکل ۲-۳: تأثیر مقیاس و جابجایی زمانی روی موجک مادر ۲۵
- شکل ۳-۳: فضاهای چند درجه‌ی تفکیک ۲۸
- شکل ۴-۳: (الف) تابع مقیاس هار. (ب) موجک هار ۳۰
- شکل ۵-۳: یک مرحله تبدیل موجک (الف) تحلیل (تجزیه) (ب) ترکیب ۳۳
- شکل ۶-۳: تبدیل موجک ۲ مرحله‌ای بر روی تصویر ۳۴
- شکل ۷-۳: (الف) هفت زیرتصویر تولید شده توسط فرآیند نشان داده شده ۳۵
- شکل ۸-۳: تبدیل به روش طرح بالابر دوبعدی ۳۶
- شکل ۱-۴: نمایش بلوکی معماری پیشنهادی ۳۹
- شکل ۲-۴: مدار مربوط به پردازنده تبدیل موجک ۵/۳ ۴۰
- شکل ۳-۴: مدار مربوط به واحد کنترل معماری پیشنهادی ۴۲

- شکل ۱-۵: شکل موج ورودی و خروجی معماری پیشنهادی ۴۹
- شکل ۲-۵: شکل موج ورودی و خروجی مدار پیشنهادی پس از گذشت نه چرخه ساعت از فعال شدن Reset ۵۰
- شکل ۳-۵: الف) تصویر ورودی ب) تبدیل موجک یک بعدی تصویر ج) تبدیل موجک دو بعدی تصویر ۵۱
- شکل ۴-۵: تبدیل موجک دو بعدی دو سطحی تصویر ۵۲
- شکل ۵-۵: الف) تصویر خروجی برنامه VHDL ب) تصویر خروجی برنامه متلب ۵۲
- شکل ۶-۵: تصویر ورودی ۵۳
- شکل ۷-۵: الف) تبدیل موجک یک بعدی تصویر ب) تبدیل موجک دو بعدی تصویر ۵۳
- شکل ۸-۵: تبدیل موجک دو بعدی دو سطحی تصویر ۵۴
- شکل ۹-۵: الف) تصویر خروجی برنامه VHDL ب) تصویر خروجی برنامه متلب ۵۴
- شکل ۱۰-۵: تصویر Lenna ۵۵
- شکل ۱۱-۵: الف) تبدیل موجک یک بعدی تصویر ب) تبدیل موجک دو بعدی تصویر ۵۵
- شکل ۱۲-۵: تبدیل موجک دو بعدی دو سطحی تصویر ۵۶
- شکل ۱۳-۵: الف) تصویر خروجی برنامه VHDL ب) تصویر خروجی برنامه متلب ۵۶

فهرست جداول

- جدول ۱-۲ : منابع سخت‌افزاری به‌کاررفته در مدل [۱] ۷
- جدول ۲-۲: منابع سخت‌افزاری به‌کاررفته در [۲] ۹
- جدول ۳-۲: منابع سخت‌افزاری استفاده شده در این معماری ۱۲
- جدول ۴-۲: منابع سخت‌افزاری به‌کاررفته در معماری مرجع [۹] ۱۵
- جدول ۵-۲: نتایج حاصل از معماری پیشنهادی مرجع [۱۰] ۱۷
- جدول ۶-۲: حداکثر و حداقل فرکانس کاری در مرجع [۱۰] ۱۷
- جدول ۱-۵: تعداد منابع سخت‌افزاری استفاده شده در معماری پیشنهادی ۵۷
- جدول ۲-۵: مقایسه نتایج معماری پیشنهادی با نتایج سایر مقالات ۵۸
- جدول ۳-۵: مقایسه حداکثر فرکانس معماری پیشنهادی با فرکانس سایر مقالات ۵۹

فصل اول: پیش‌گفتار

۱-۱ تعریف مساله و هدف از تحقیق

در سال‌های اخیر تبدیل موجک به ابزاری کارآمد جهت کاربردهای مختلف در پردازش سیگنال تبدیل شده است. برتری این تبدیل نسبت به تبدیل‌های دیگر بیشتر به این دلیل است که یک تحلیل با دقت چند درجه تفکیک^۱ بر روی سیگنال انجام می‌دهد که قابلیت تنظیم دقت در هر دو حوزه زمان و فرکانس را دارا است. تبدیل موجک به طور گسترده‌ای در بسیاری از زمینه‌های پردازش صدا و ویدئو مانند حذف نویز تصاویر، شناسایی الگو و کدگذاری و فشرده‌سازی تصویر مورد استفاده قرار می‌گیرد. به همین دلیل ارائه راهکارهایی که سبب افزایش سرعت پیاده‌سازی تبدیل موجک می‌شود مورد توجه قرار گرفته است.

جهت افزایش سرعت یکی از بهترین راهکارها استفاده از پردازش موازی است و استفاده از تراشه‌های FPGA راهکار مناسبی جهت دستیابی به پردازش موازی است که قابلیت انعطاف و توانایی موازی‌سازی بالایی دارند و با استفاده از آنها می‌توان با یک طراحی مناسب، سرعت و کارایی بالایی برای پیاده‌سازی تبدیل موجک فراهم نمود.

برای اعمال تبدیل موجک گسسته بر روی یک سیگنال از دو روش اساسی استفاده می‌شود:

روش اول بانک‌های فیلتری^۲ است که در این روش سیگنال ورودی با عبور همزمان از یک فیلتر بالاگذر و یک فیلتر پایین‌گذر و سپس کاهش نرخ نمونه‌برداری در حوزه فرکانس به دو مؤلفه سیگنال تقریب و سیگنال جزئیات تقسیم می‌شود. پیاده‌سازی این روش بر پایه کانولوشن نمونه‌های سیگنال ورودی با ضرایب فیلتر است.

روش دوم روش طرح بالابر^۳ است که در این روش سیگنال ورودی ابتدا به دو قسمت زوج و فرد تقسیم می‌شود. سپس با استفاده از بلوک‌های پیش‌بینی^۴ و به‌روزرسانی^۵، ضرایب تقریب و جزئیات

¹ Multiresolution Decomposition

² Filter bank

³ Lifting scheme

⁴ predict

⁵ update

سیگنال محاسبه می‌شوند. از مزایای این روش نسبت به روش بانک‌های فیلتری می‌توان به حجم محاسبات و میزان حافظه کم‌تر اشاره کرد.

اعمال تبدیل موجک گسسته روی تصاویر (سیگنال‌های دوبعدی) با استفاده از روش‌های بیان‌شده در دو پاراگراف قبل به این صورت انجام می‌گیرد که ابتدا عملیات تبدیل موجک بر روی تک‌تک سطرهای تصویر انجام می‌شود و سپس از تصاویر خروجی به دست آمده در راستای ستون‌ها تبدیل موجک گرفته می‌شود. در فصل سوم به صورت کامل در مورد مبانی تبدیل موجک توضیح داده می‌شود.

در این پایان‌نامه به ارائه یک معماری برای پیاده‌سازی تبدیل موجک گسسته دو بعدی مبتنی بر روش طرح بالابر با استفاده از موجک $5/3$ جهت کاربرد در فشرده‌سازی بدون اتلاف تصویر پرداخته شده است که هدف از انجام این کار کاهش پیچیدگی محاسبات و همچنین کاهش منابع سخت‌افزاری مورد استفاده برای پیاده‌سازی و افزایش فرکانس کاری است. این معماری شامل یک واحد پردازنده جهت محاسبه ضرایب موجک، یک واحد حافظه خارجی جهت ذخیره ضرایب موجک و یک واحد کنترل جهت کنترل جریان داده در پردازنده و همچنین تولید خطوط آدرس حافظه برای ذخیره داده در حافظه خارجی است.

۲-۱ مروری بر فصل‌های پایان‌نامه

فصل‌های دیگر این پایان‌نامه به این صورت است:

در فصل دوم به بررسی مقالات و پژوهش‌های انجام‌شده در زمینه پیاده‌سازی تبدیل موجک گسسته در FPGA و معماری‌های پیشنهادشده توسط آن‌ها به صورت مختصر می‌پردازیم.

در فصل سوم به توضیح تاریخچه و مفاهیم کدگذاری زیر بانندی و مبانی تبدیل موجک و روابط و فرمول‌های مربوط به آن می‌پردازیم. در فصل چهارم معماری پیشنهادی ارائه‌شده توسط این پایان‌نامه برای پیاده‌سازی تبدیل موجک را بیان می‌کنیم و قسمت‌های مختلف این معماری را توضیح می‌دهیم.

در فصل پنجم نتایج شبیه‌سازی مربوط به معماری پیشنهادی با نرم‌افزار Xilinx Ise بیان می‌شود و سپس معماری را با تصاویر ورودی مختلف ارزیابی کرده و نتایج حاصل را با نتایج پیاده‌سازی تبدیل موجک در محیط نرم‌افزار متلب و با دیگر معماری‌های ارائه‌شده در مقالات مختلف مقایسه می‌کنیم.

در فصل ششم نیز به نتیجه‌گیری از پژوهش انجام‌شده و ارائه پیشنهادهایی برای توسعه کار

پژوهش

می‌پردازیم.

فصل دوم: مروری بر پژوهش‌های

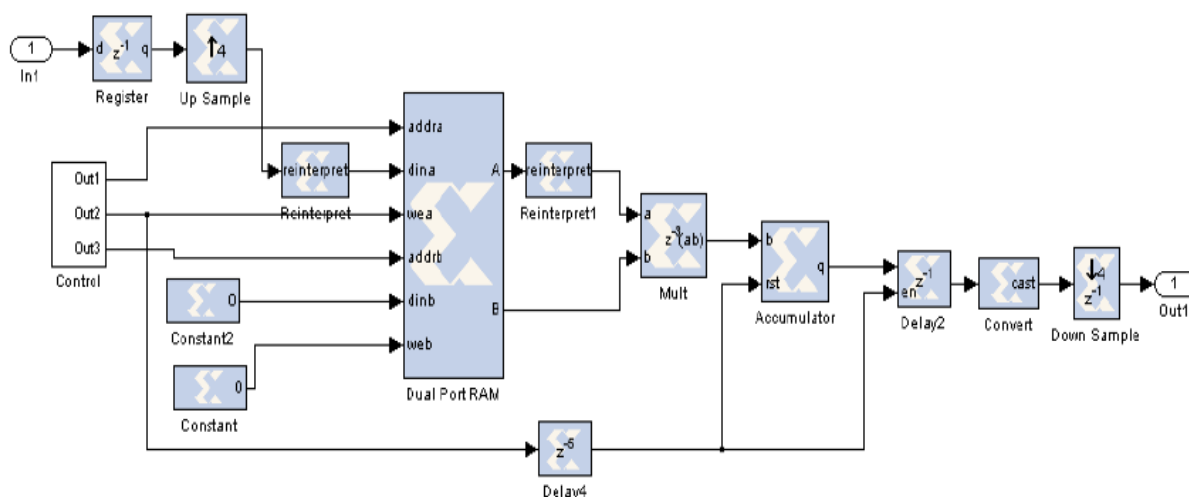
گذشته

۱-۲ مقدمه

تاکنون پژوهش‌های زیادی در زمینه پیاده‌سازی تبدیل موجک گسسته در FPGA انجام شده است. هر یک از این پژوهش‌ها با ارائه روش و یا مدلی برای پیاده‌سازی تبدیل موجک پرداخته‌اند که در این فصل به صورت مختصر به توضیح بعضی از کارهای انجام شده می‌پردازیم.

۲-۲ پژوهش‌های گذشته

در [۱] جنبه‌های عملی و نظری در ارتباط با پیاده‌سازی سخت‌افزاری تبدیل موجک ارائه شده است. این مقاله پاسخ‌های مختلف را برای پیاده‌سازی سخت‌افزاری تبدیل موجک در FPGA با استفاده از محیط سیمولینک متلب مورد بررسی قرار داده است. از موجک دابیشز ۴ برای پیاده‌سازی استفاده شده است و بلوک‌های اصلی به کاررفته در آن شامل یک حافظه Dual Ram برای ذخیره‌سازی ضرایب موجک و داده، یک ضرب کننده، دو بافر و یک بلوک کنترل می‌باشد. در شکل ۱-۲ مدل ارائه شده توسط این مقاله مشاهده می‌شود.



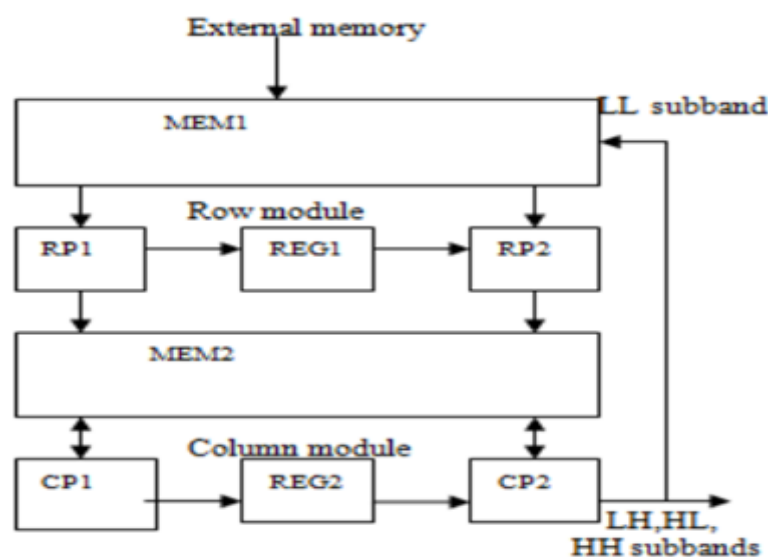
شکل ۱-۲: مدل ارائه شده برای پیاده‌سازی تبدیل موجک در [۱]

منابع سخت‌افزاری به کاررفته در این معماری به صورت جدول ۱-۲ است:

جدول ۱-۲: منابع سخت‌افزاری به‌کاررفته در مرجع [۱]

Resources	Spartan3 HW multipliers	%	Spartan3	%
Number of slices	۱۹۸۰	۴۲.۵	۳۴۴۰	۷۳.۸
Number of Slice Flip Flops	۲۸۸۴	۳۰.۹	۵۴۳۶	۵۸.۳
Block RAM	۱۲	-	۱۲	-
LUT	۱۹۵۲	۲۰.۹	۵۱۰۴	۵۴.۸
Multiplier	۱۲.۲۰	۸۰	۰	۰

در [۲] یک معماری برای پیاده‌سازی تبدیل موجک گسسته برای فشرده‌سازی تصویر ارائه شده است. این طراحی از استاندارد Jpeg 2000 تبعیت می‌کند و می‌تواند برای هر دو فشرده‌سازی با اتلاف و بی‌اتلاف استفاده شود. از روش طرح بالابر برای این پیاده‌سازی استفاده شده است و همچنین این طراحی می‌تواند برای فشرده‌سازی در یک سیستم روباتیک مورد استفاده قرار گیرد. در شکل ۲-۲ نمایش بلوکی یک مرحله از طرح بالابر با استفاده از معماری ارائه‌شده در این مقاله را نمایش می‌دهد.

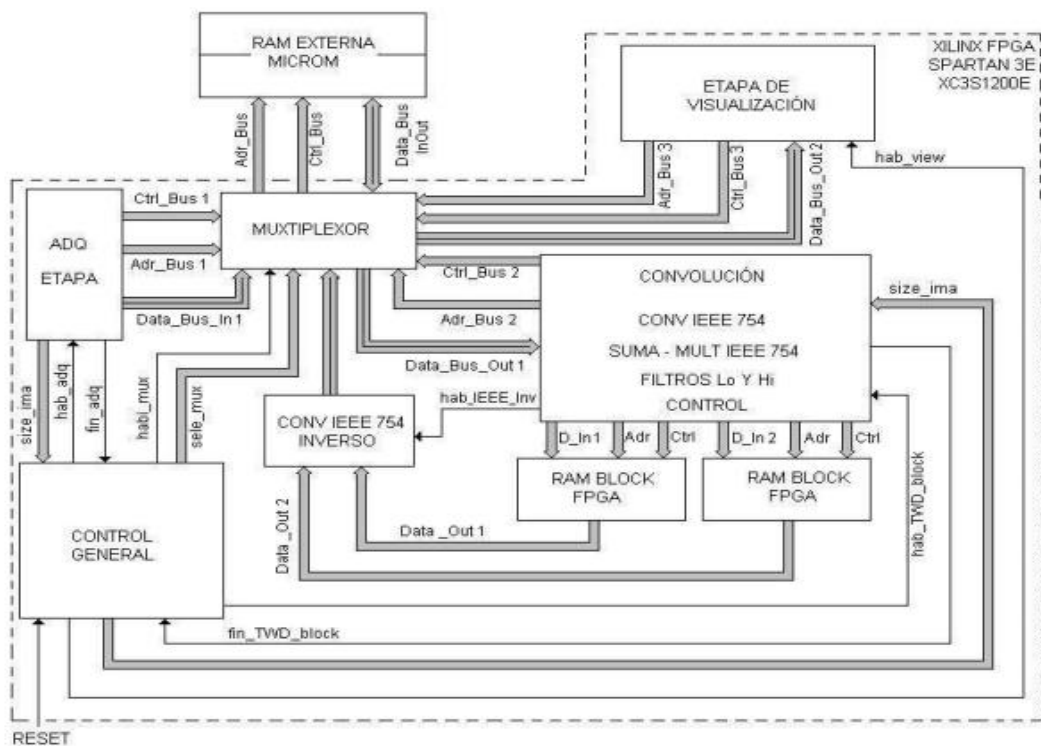


شکل ۲-۲: نمایش بلوکی یک مرحله از طرح بالا بر با استفاده از معماری ارائه شده در [۲]

این نمایش بلوکی شامل بخش‌های محاسباتی سطر و ستون و دو واحد حافظه (MEM1 و MEM2) است. بخش سطر شامل دو پردازنده RP1 و RP2 با یک ثبت REG1 در بین آنها است و بخش ستون نیز شامل دو پردازنده CP1 و CP2 با یک ثبت REG2 در میان آنها می‌باشد. هر یک از این چهار پردازنده از دو جمع کننده، یک ضرب کننده و یک شیفتردهنده تشکیل می‌شود. CP1 و RP1 فیلترهای پیش‌بینی و RP2 و CP2 فیلترهای به‌روز رسانی هستند. این طراحی می‌تواند اندازه حافظه داخلی را کاهش و توان عملیاتی را بهبود ببخشد.

مرجع [۳] با استفاده از فیلتر خانواده دابیشز $2(db)$ تبدیل موجک گسسته را در دو بعد در FPGA پیاده‌سازی کرده است. معماری پیشنهاد شده در این مقاله با تصاویر ورودی مختلف ارزیابی شده است و هدف از این طراحی توسعه سیستم‌های سخت‌افزاری و رمزدار کردن تصاویر در آینده با استفاده از تبدیل موجک برای اطلاعات امنیتی بیان شده است. همچنین از روش بانک فیلتری برای طراحی معماری در این مقاله استفاده شده است.

معماری ارائه شده در شکل ۲-۳ نشان داده شده است:



شکل ۳-۲: معماری ارائه شده توسط مرجع [۳]

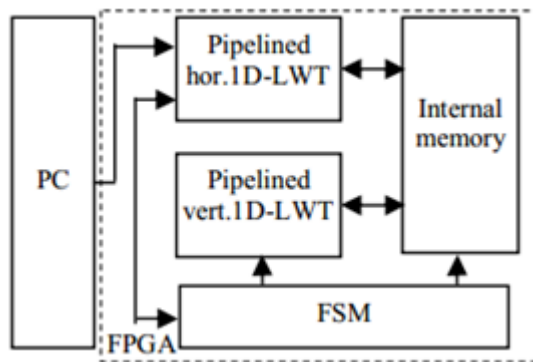
منابع سخت‌افزاری به کار رفته در این مقاله نیز در جدول ۲-۲ نشان داده شده است:

جدول ۲-۲: منابع سخت‌افزاری به کار رفته در [۳]

Hardware Resources	استفاده شده	در دسترس	%
Number of slices	۳۸۴۲	۸۶۷۲	٪۴۴.۳
Number of Slice Flip Flops	۱۶۰۸	۱۷۳۴۴	٪۹.۳
Number of 4input LUTs	۶۵۸۹	۱۷۳۴۴	٪۴۴.۳
Number of bonded IOBs	۱۰	۲۴	٪۴۴.۳
Clock Rate			۵۰ MHz

در [۴] یک معماری VLSI با سرعت بالا برای پیاده‌سازی تبدیل موجک گسسته دوبعدی ارائه شده است که بر اساس فرمول‌های ریاضی تبدیل موجک دوبعدی است. در این مقاله از یک طرح خط لوله^۱ که سبب می‌شود یک مسیر بحرانی فقط به اندازه یک زمان تأخیر جمع کننده طول بکشد جهت افزایش نرخ چرخه ساعت استفاده شده است. این طراحی به ازای k سطح تجزیه برای یک تصویر $N \times N$ به $N/2 \times N/2$ چرخه ساعت نیاز دارد و در یک پایگاه زمان حقیقی از یک سنسور تصویر CMOS تطبیق داده شده است.

در [۵] روشی برای پیاده‌سازی تبدیل موجک گسسته دوبعدی با طرح بالابر برای فشردگی با اتلاف تصویر بر روی FPGA ارائه شده است که بر مبنای فیلترهای دابیشز $9/7$ می‌باشد. معماری ارائه‌شده در شکل ۴-۲ قابل مشاهده است.



شکل ۴-۲: مدل ارائه شده توسط مرجع [۵]

معماری ارائه‌شده شامل دو واحد DWT^2 یک بعدی برای تبدیل موجک در راستای افقی و عمودی، یک واحد کنترل و یک بلوک حافظه داخلی است. جهت پردازش یک زیرتصویر همه سطرها به گذرگاه PCI در FPGA فرستاده می‌شوند و سپس به واحد 1D-LWT افقی با استفاده از خط لوله منتقل می‌شوند. ضرایب محاسبه‌شده در این بخش در حافظه داخلی ذخیره می‌شوند. سپس با استفاده از واحد 1D-LWT عمودی تبدیل موجک در راستای عمودی گرفته می‌شود. واحد کنترل نیز وظیفه

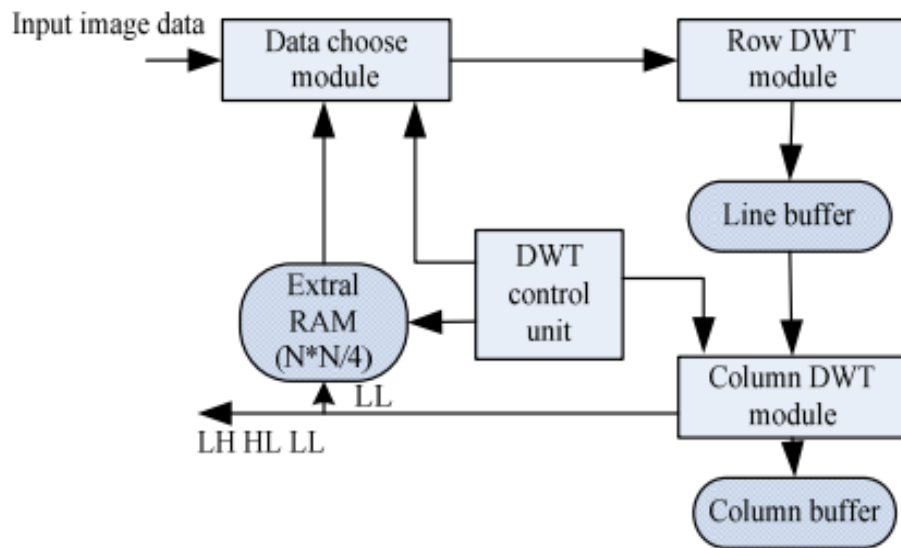
¹ pipeline

² Discrete wavelet transform

تولید سیگنال فعال کننده و خطوط آدرس را بر عهده دارد.

در [۶] نیز روشی برای پیاده‌سازی تبدیل موجک گسسته دوبعدی بر اساس طرح بالابر ارائه شده

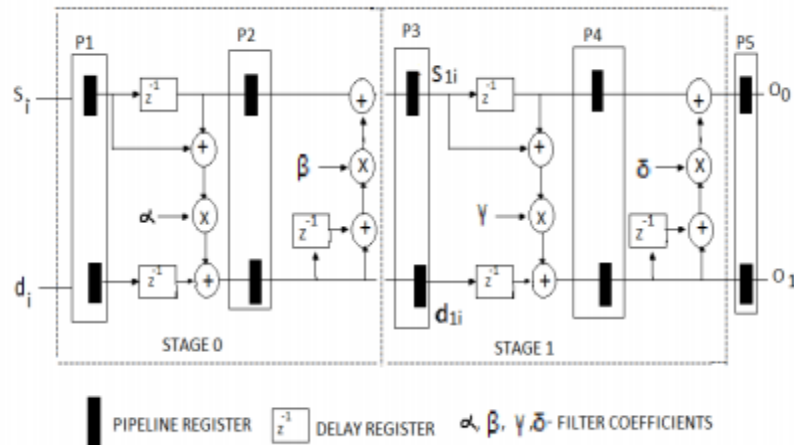
است که در شکل ۲-۵ مشاهده می‌شود:



شکل ۲-۵: مدل پیشنهادی تبدیل موجک دوبعدی مرجع [۶]

این معماری شامل پنج بخش^۱ است: بخش انتخاب داده، بخش تبدیل موجک در راستای سطر، بخش تبدیل موجک در راستای ستون و بخش‌های واحد کنترل و حافظه خارجی. حافظه خارجی که اندازه آن $N^2/4$ است جهت ذخیره‌سازی ضرایب باند تقریب (LL) استفاده می‌شود که N تعداد سطرها و ستون‌های تصویر ورودی است. واحد کنترل DWT نیز ترتیب زمانی کل سیستم را کنترل می‌کند. در [۷] روشی برای طراحی معماری خط لوله برای انجام محاسبات تبدیل موجک گسسته یک بعدی ممیز ثابت با استفاده از طرح بالابر ارائه شده است که تمرکز اصلی آن بر روی کاهش تعداد چرخه‌های ساعت و منابع سخت‌افزاری است. نمایش ممیز ثابت به منابع سخت‌افزاری کمتری در مقایسه با نمایش ممیز شناور دارد. در طراحی این معماری از موجک ۹/۷ استفاده شده است. معماری ارائه‌شده به صورت شکل ۲-۶ است:

¹ module



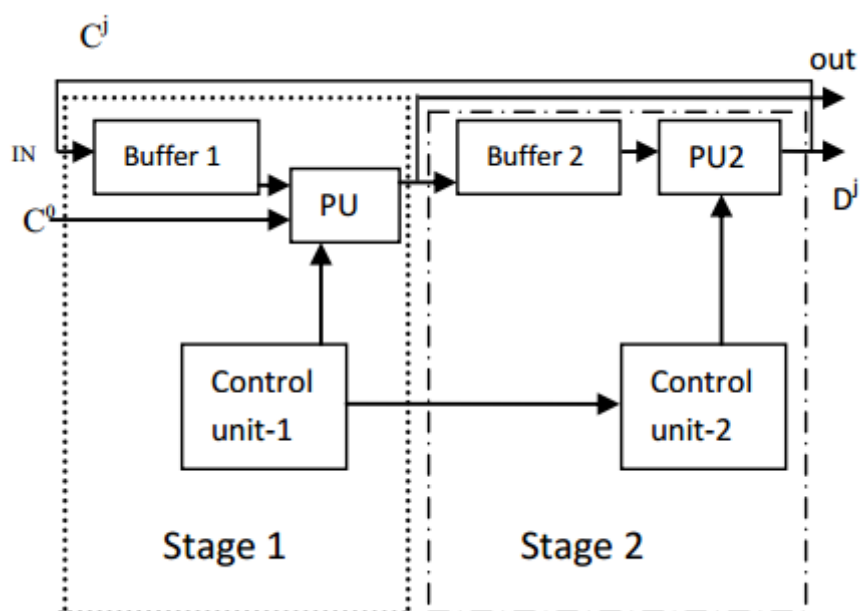
شکل ۲-۶: مدل تبدیل موجک یک بعدی پیشنهادی مرجع [۷]

در این معماری، پنج طبقه خط لوله برای انجام محاسبات تبدیل موجک یک بعدی طراحی شده‌اند که همه این طبقات نیاز به اشتراک‌گذاری محاسبات دارند؛ بنابراین همه طبقات خط لوله به صورت همزمان با یکدیگر کار می‌کنند و ثبات‌های خط لوله نیز در این معماری در مکان‌های مناسب برای بهینه‌سازی منابع سخت‌افزاری قرار داده شده‌اند. منابع سخت‌افزاری استفاده شده در این معماری به صورت جدول ۲-۳ است:

جدول ۲-۳: منابع سخت‌افزاری استفاده شده در معماری [۷]

Resource	استفاده شده	در دسترس	%
CLB slices	۱۵۸	۴۹۲۸	٪۳
Number of Slice Flip Flops	۲	۹۸۵۶	٪۲
Number of 4input LUTs	۱۳۳	۹۸۵۶	٪۱
Number of bonded IOBs	۷۰	۲۴۸	٪۲۸

در [۸] یک معماری VLSI خط لوله برای پیاده‌سازی تبدیل موجک گسسته دوبعدی با استفاده از فیلتر دایبیشز ۲ ارائه شده است که تمرکز اصلی آن بر روی کاهش تعداد چرخه‌های ساعت مورد نیاز برای انجام محاسبات است. در این معماری از دو طبقه خط لوله استفاده شده است که هر کدام از آنها از واحدهای پردازشی یکسانی برخوردار هستند و همزمان با هم به انجام محاسبات مربوط به تبدیل موجک گسسته می‌پردازند؛ اما واحدهای کنترل آن‌ها با هم تفاوت دارد زیرا طبقه ۱ محاسبات مربوط به سطرها (افقی) و طبقه ۲ به صورت همزمان محاسبات مربوط به ستون‌های تصویر را انجام می‌دهد. به منظور همزمان‌سازی عملیات مربوط به طبقه ۲ با طبقه ۱ از یک بافر استفاده شده که نمونه‌های خروجی پایین‌گذر از طبقه ۱ را ذخیره می‌کند و جهت همزمان‌سازی بین نتایج سطوح تجزیه یک ثبات ۱۶ بیتی به عنوان بافر در ورودی طبقه ۱ قرار داده شده است. شکل ۲-۷ نمایش بلوکی ترکیب همه منابع مورد نیاز برای طراحی این معماری را نشان می‌دهد.

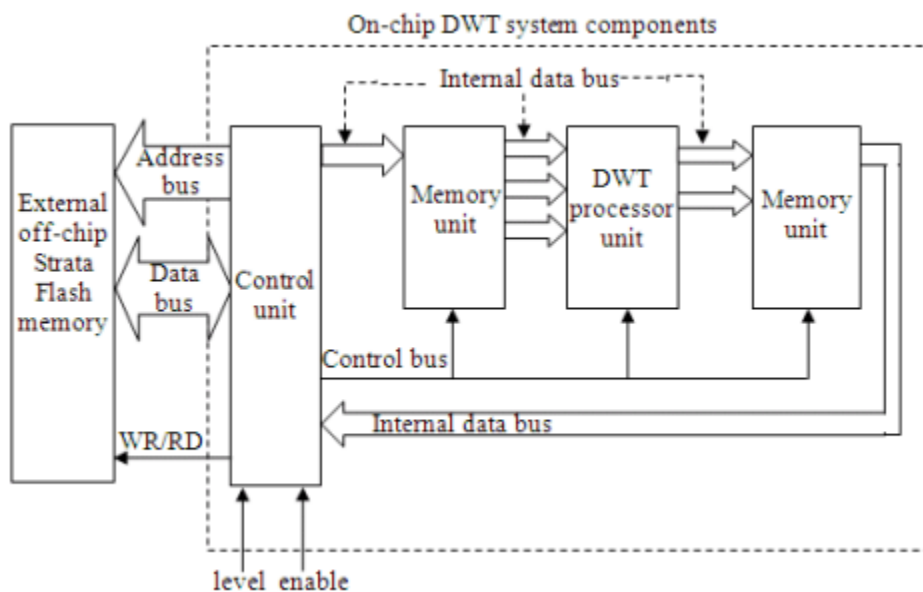


شکل ۲-۷: نمایش بلوکی مدل پیشنهادی مرجع [۸]

در این معماری دو واحد پردازشی به نام‌های PU در طبقه ۱ و PU2 در طبقه ۲ قرار داده شده‌اند.

فرآیند همزمان‌سازی داده‌های خروجی با معرفی یک بافر در طبقه ۱ و یک بافر در طبقه ۲ انجام می‌یابد. بافر طبقه ۱ خروجی هر سطح از تجزیه را ذخیره و آن را به عنوان ورودی برای سطح بعدی آماده می‌کند. در حالی که بافر طبقه ۲ داده‌های خروجی از طبقه ۱ را ذخیره و به عنوان داده‌های ورودی طبقه ۲ آماده می‌کند. در این معماری از چهار ضرب کننده و دو جمع کننده استفاده شده است.

در [۹] یک معماری برای پیاده‌سازی تبدیل موجک دوبعدی جهت کاربرد در فشرده‌سازی تصویر ارائه شده است که معماری ارائه شده بر اساس طرح بالا بر می‌باشد و از موجک $5/3$ جهت پیاده‌سازی استفاده شده است. این معماری شامل یک واحد کنترل، یک واحد پردازنده، دو حافظه داخلی روی تراشه و یک حافظه خارجی بیرون تراشه است. نمایش بلوکی معماری ارائه شده در شکل ۸-۲ مشاهده می‌شود.



شکل ۸-۲: نمایش بلوکی معماری ارائه شده در [۹]

در معماری شکل ۸-۲ از دو واحد حافظه روی تراشه استفاده شده است که برای یک تصویر $N*N$

اندازه هر کدام از آن‌ها 2N است. واحد حافظه اول مانند یک طبقه برش^۱ برای طرح بالابرد عمل می‌کند و پیکسل‌های زوج را از پیکسل‌های فرد جدا می‌کند. واحد پردازشگر DWT نیز عملیات پیش‌بینی و به روزرسانی را انجام می‌دهد. در طراحی این واحد از سه ثبات جهت ذخیره‌سازی داده‌هایی که از حافظه داخلی اول می‌آیند و چهار جمع‌کننده و دو شیفت‌دهنده استفاده شده است. واحد کنترل دو وظیفه بر عهده دارد: ۱- کنترل اجزای سیستم DWT روی تراشه و همچنین کنترل جریان داده در معماری ارائه شده را بر عهده دارد. ۲- تعامل و هماهنگی بین سیگنال‌ها و گذرگاه‌ها با حافظه خارجی را فراهم می‌کند.

حافظه خارجی نیز برای ذخیره‌سازی تصویر اصلی و سپس ضرایب DWT به کار می‌رود. منابع سخت‌افزاری به کار رفته در این معماری در جدول ۴-۲ مشاهده می‌شود:

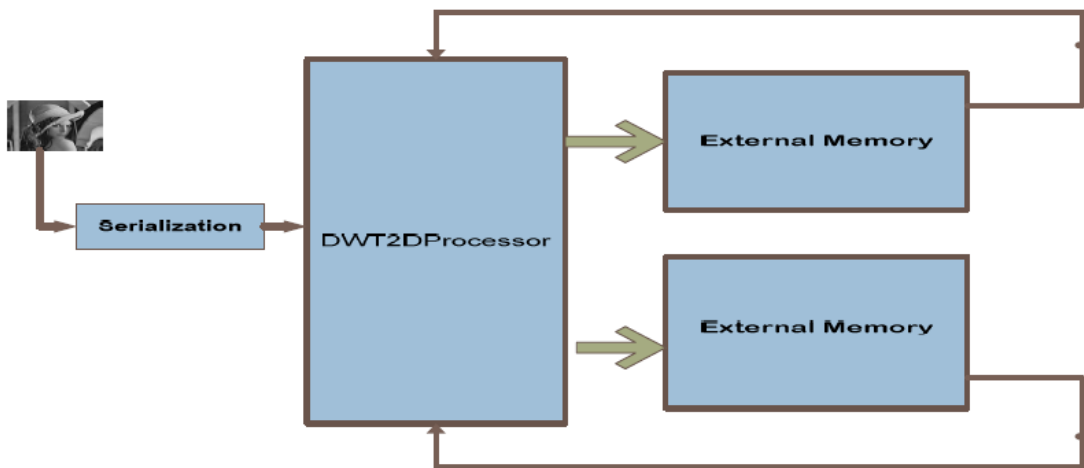
جدول ۴-۲: منابع سخت‌افزاری به کار رفته در معماری مرجع [۹]

	استفاده شده	در دسترس	%
Number of slices	۷۸	۴۶۵۶	٪۱
Number of Slice Flip Flops	۱۱۹	۹۳۱۲	٪۱

Maximum Frequency	۶۲.۷۹۷ MHZ
--------------------------	------------

در [۱۰] روشی برای پیاده‌سازی بلادرنگ تبدیل موجک گسسته دو بعدی با موجک ۵/۳ در FPGA ارائه شده است که از محیط سیمولینک نرم‌افزار Xilinx System Generator برای شبیه‌سازی استفاده شده است. معماری ارائه شده در شکل ۲-۹ نشان داده شده است:

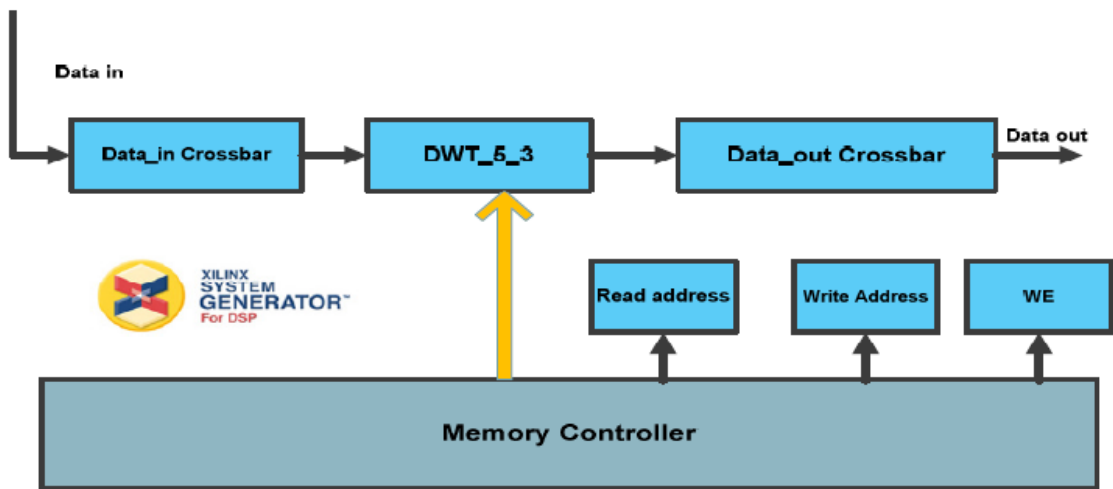
¹ Split



شکل ۹-۲: معماری پیشنهادی در مرجع [۱۰]

با توجه به شکل ۹-۲ تصویر ورودی پس از آنکه به صورت یک بردار درآمد است وارد پردازنده می شود و پس از انجام عملیات تبدیل موجک در حافظه های خارجی ذخیره می شود. ضرایب تقریب در یک حافظه و ضرایب جزییات در حافظه دیگر ذخیره می شوند و سپس این ضرایب بار دیگر وارد پردازنده می شوند تا دوباره از آنها تبدیل موجک گرفته شود و در نهایت نتایج نهایی در حافظه ها ذخیره می شوند.

مدار مربوط به پردازنده در شکل ۱۰-۲ نشان داده شده است:



شکل ۱۰-۲: مدار مربوط به پردازنده در معماری پیشنهادی مرجع [۱۰]

با توجه به شکل ۲-۱۰ مدار پردازنده شامل یک واحد کنترل حافظه برای خواندن و نوشتن آدرس ضرایب، یک واحد DWT که عملیات تبدیل موجک ۵/۳ را انجام می‌دهد و دو واحد Crossbars که برای جداسازی پیکسل‌های تصویر از یکدیگر استفاده می‌شود. نتایج حاصل از این معماری به صورت جدول ۲-۵ است:

جدول ۲-۵: نتایج حاصل از معماری پیشنهادی مرجع [۱۰]

	استفاده شده	در دسترس	%
Number of slices	۲۷۲	۲۳۸۷۲	٪۱.۱۳
Number of Slice Flip Flops	۴۹۷	۴۷۷۴۴	٪۱.۰۴
Number of 4input LUTs	۳۴۴	۴۷۷۴۴	٪۰.۷
Number of bonded IOBs	۹۱	۴۶۹	٪۱۹
Number of GCLKs	۱	۲۴	٪۴

حداکثر و حداقل فرکانس کاری نیز به صورت جدول ۲-۶ است:

جدول ۲-۶: حداکثر و حداقل فرکانس کاری در مرجع [۱۰]

Minimum Periods	۴.۸۶۳ ns
Maximum Frequency	۲۰.۵.۴۲۳ MHZ

فصل سوم: مبانی تئوری تبدیل موجک

در این فصل به توضیح تاریخچه و مفاهیم کدگذاری زیرباندی و مبانی تبدیل موجک و روابط و فرمول‌های آن و کاربرد آن در فشرده‌سازی تصویر می‌پردازیم.

۲-۳ چرا تبدیل موجک؟

پیش از بیان مفاهیم تبدیل موجک و کاربرد آن در فشرده‌سازی تصویر لازم است به یک سؤال پاسخ داده شود:

مشکل تبدیل^۱ DCT چیست و چرا تبدیل موجک جایگزین آن شده است؟

تبدیل DCT و سایر تبدیلاتی که بر اساس بلوک‌بندی هستند، به این شکل عمل می‌کنند که تصویر را به تعدادی بلوک‌های ناهمپوشان تقسیم و هر بلوک را به صورت مجزا پردازش می‌کنند. در نرخ بیت‌های بسیار کوچک لازم است که ضرایب تبدیل با گام بزرگی کوانتیزه شوند و بنابراین در حین کدگذاری، خطای بازسازی بزرگ و تصویر بازسازی‌شده نامطلوب می‌شود. این خطا در مرز بلوک‌ها یا لبه‌ها مشهودتر از بقیه نواحی بوده و یک اثر نامطلوب به نام اثر بلوکی^۲ تولید می‌کند. یکی از راه‌ها جهت حذف یا کاهش این اثر نامطلوب این است بلوک‌ها را به صورت همپوشان انتخاب کنیم. روش دوم، تبدیل متعامد همپوشان (LOT)^۳ [11] است که تبدیل موجک حالت خاصی از این تبدیل است و بنابراین انتظار می‌رود اثرات بلوکی از خود نشان ندهد.

تبدیل DCT در استاندارد JPEG مورد استفاده قرار گرفت زیرا در آن زمان هنوز روش موثری برای کدگذاری ضرایب تبدیل موجک در مقایسه با تبدیل DCT وجود نداشت؛ اما در JPEG2000 از روش تبدیل موجک استفاده شده است.

از آن زمان تاکنون روش‌های فشرده‌سازی تصویر مبتنی بر تبدیل موجک رشد و پیشرفت فراوان و

¹ Discrete cosine transform

² Blocking Artifact

³ Lapped Orthogonal Transform

چشمگیری داشته است که بیشترین سهم در این زمینه را باید به آقای ژوزف شاپیرو^۱ داد که روش موجک درخت صفر جاسازی شده EZW^۲ را ارائه داد و پیشرفت بزرگی در زمینه کدگذاری ضرایب موجک ایجاد کرد [12].

۳-۳ کدگذاری زیرباندی

کدگذاری زیرباندی اولین بار در سال ۱۹۷۶ [۱۳] پیشنهاد گردید و تاکنون قدرت و کارایی آن در زمینه کدگذاری صوت و تصویر به عنوان یک ابزار ساده و قدرتمند به اثبات رسیده است. ایده اصلی این تکنیک، بخش‌بندی طیف سیگنال به چندین محدوده یا باند فرکانسی^۳ و سپس کدگذاری و انتقال هر باند به طور مجزا است. این کار مختص کدگذاری تصویر ارائه و استفاده گردید. در توجیه این کار ذکر سه نکته قابل تأمل است. اول این که طیف تصاویر طبیعی اغلب طیفی غیرخطی است و بیشتر انرژی این‌گونه تصاویر در محدوده فرکانس‌های پایین متمرکز است. نکته دوم این که میزان درک انسان از نویز در هر دو باند فرکانسی بالا و پایین افت می‌کند؛ بنابراین طراح می‌تواند سیستم فشرده‌سازی را به گونه‌ای طراحی کند که مقدار اعوجاج حاصل از فشرده‌سازی بر طبق یک معیار ادراکی تنظیم شود. بالأخره، سوم این که از آنجا که در این روش، تصاویر به صورت کلی و یکجا پردازش می‌شوند نه به صورت بلوکی، بنابراین مساله‌ای به نام اعوجاج بلوکی مشابه با آن چه در مورد روش‌های مبتنی بر تبدیل بلوکی، مانند DCT مطرح بود وجود نخواهد داشت.

بنابراین اگرچه ایده زیرباندی مشابه با تبدیل فوریه، مبتنی بر تحلیل فرکانسی تصویر است اما بانک‌های فیلتری آن خاصیت ناهمبسته‌سازی بهتری دارند که این خاصیت در مورد تصاویر طبیعی مناسب‌تر و مطلوب‌تر است. برای توضیح بیشتر به این نکته توجه کنید که توابع پایه فوریه در حوزه فرکانسی کاملاً دقیق اما در حوزه مکانی دارای دقت نمی‌باشند؛ به عبارت دیگر انرژی آن‌ها در سرتاسر

¹ Juseff Shapiro

² Embedded Zerotree Wavelet

³ Frequency Band

بازه مکانی گسترده شده است. این امر در صورتی که پیکسل‌های تصویر همواره و همیشه باهم همبستگی داشته باشند، مشکلی محسوب نمی‌شود اما در عمل چنین نبوده و پیکسل‌های تصویر در بازه بسیار محدودی باهم همبستگی دارند. به ویژه در نقاط ناپیوستگی تصاویر مانند لبه‌ها، بازه همبستگی مذکور بسیار کوتاه است. بر خلاف توابع پایه فوریه، توابع پایه‌ی زیرباندی نه تنها تمرکز (یا دقت) فرکانسی^۱ نسبتاً خوبی دارند، بلکه در حوزه مکانی نیز فشرده و متمرکز می‌باشند. اگر لبه‌های تصویر بیش از حد به هم نزدیک نباشند، اغلب توابع یا عناصر پایه‌ی زیرباندی با این نواحی مرز مشترک نداشته و بنابراین به طور متوسط خاصیت ناهمبسته‌سازی خوبی از خود نشان می‌دهند (توجه شود که انجام فشرده‌سازی مستلزم حذف یا کاهش تزیاید بوده و کاهش تزیاید نیز مستلزم ناهمبسته‌سازی است).

در کدگذاری زیرباندی، جداسازی باندها از یکدیگر توسط فیلتر کردن تصویر ورودی به کمک بانکی از فیلترهای تحلیل میان‌گذر، مطابق با شکل ۳-۱، انجام می‌شود. در این شکل ملاحظه می‌کنید که جهت هماهنگ کردن طیف تصاویر تجزیه‌شده‌ی خروجی با مشخصات سیستم بینایی انسان، فیلترها به صورت باندهای اکتاو^۲ (یعنی افزایش عرض باندها به صورت توان صحیحی از ۲) طراحی و تنظیم شده‌اند.

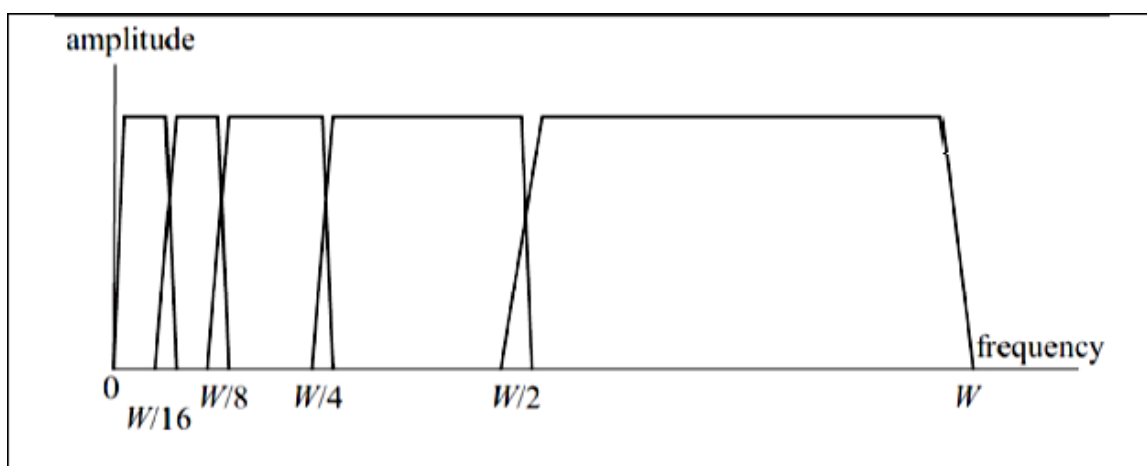
از آنجا که پهنای باند هر نسخه از تصویر فیلتر شده کاهش یافته است، از نظر تئوری می‌توان با نرخ کمتری (بر طبق قضیه نایکوئیست) نمونه‌برداری کرد و تعدادی زیرتصویر^۳ کاهش بعد یافته به دست آورد. سپس این زیرتصویرها کوانتیزه، کدگذاری و ارسال می‌شوند. در واحد کدگشا، زیرتصویرهای دریافتی به ابعاد اولیه بازگردانده شده و از مجموعه‌ای از فیلترها به نام بانک فیلتری ترکیب عبور داده می‌شوند. این بانک فیلتری عمل درونیابی و ترکیب زیرتصویرها به منظور بازسازی تصویر اولیه را انجام می‌دهد.

¹ Frequency Concentration

² Octave

³ Subimage

فیلترهایی که در عمل در بانک فیلتر استفاده می‌شوند، مانند شکل ۳-۱، ناحیه یا باند گذر محدود و غیر صفر دارند لذا در این بانک‌های فیلتری در فرآیندهای زیرنمونه‌برداری و بالا نمونه‌برداری مقداری اعوجاج همپوشانی^۱ در تصویر بازسازی شده وجود خواهد داشت. برای رفع مشکل اعوجاج، فیلترهای مورد استفاده باید رابطه خاصی با هم داشته باشند طوری که مؤلفه‌های همپوشان اثر یکدیگر را خنثی کرده و تصویری بدون اعوجاج همپوشانی حاصل شود [۱۳].



شکل ۳-۱: یک نمونه بانک‌های فیلتری میان‌گذر

۴-۳ تبدیل موجک

تبدیل موجک حالت خاصی از کدگذاری زیرباندی است که در زمینه کدگذاری تصویر و ویدیو بسیار فراگیر و متداول شده است. کدگذاری زیرباندی تصاویر مبتنی بر تحلیل فرکانسی است حال آنکه تبدیل موجک بر تئوری تقریب^۲ استوار است. با این حال از آنجا که تصاویر طبیعی به طور محلی هموار بوده و می‌توان آن را با تقریب تکه‌ای - چندجمله‌ای مدل کرد، اگر از تابع چند جمله‌ای مناسب استفاده کنیم، عمل تقریب متناظر با نوعی تحلیل فرکانسی خواهد شد که مشابه با ایده زیرباندی است. در حقیقت، تحلیل موجک راه و ابزار بسیار موثری برای تقریب چنین توابعی در اختیار ما قرار

¹ Aliasing Distortion

² Approximation Theory

می‌دهد طوری که از تعداد بسیار کمی عناصر پایه (برای انجام تقریب) استفاده می‌کند.

از نظر ریاضی، تبدیل موجک یک تابع مجذور-انتگرال پذیر $x(t)$ معادل با تجزیه این تابع بر حسب مجموعه‌ای از توابع پایه به صورت زیر است:

$$X_w(a, b) = \int_{-\infty}^{\infty} x(t) \Psi_{a,b}(t) dt \quad (1-3)$$

که در آن $\Psi_{a,b}(t)$ به نام تابع پایه شناخته می‌شود. این تابع نسخه‌ای کشیده شده و جابجا شده از یک سیگنال میان‌گذر مانند $\Psi(t)$ است که تابع موجک مادر^۱ نامیده شده و به صورت زیر تعریف می‌شود:

$$\Psi_{a,b} = \frac{1}{\sqrt{a}} \Psi\left(\frac{t-b}{a}\right) \quad (2-3)$$

که در آن a و b به ترتیب پارامترهای مقیاس^۲ و جابجایی^۳ نامیده می‌شوند. تأثیر این پارامترها در شکل ۲-۳ نشان داده شده است.

عرض تابع پایه با تغییر فاکتور مقیاس a به این شکل تغییر می‌کند که هرچه این فاکتور بزرگتر شود، عرض زمانی تابع پایه نیز بزرگتر می‌شود و بنابراین، عرض (یا پهنای باند) فرکانسی آن کمتر می‌شود؛ بنابراین به کمک این پارامتر می‌توانیم درجه تفکیک زمانی و فرکانسی را (با مصالحه‌ای که بین یکدیگر دارند) در تبدیل موجک تغییر دهیم و همین ویژگی جالب تبدیل موجک است که باعث می‌شود این تبدیل در تحلیل سیگنال‌هایی که ویژگی‌هایی با اندازه‌های مختلف دارند، مانند تصاویر طبیعی^۴، مناسب باشد. متناظر با هر اندازه‌ی ویژگی‌ای یک تابع پایه $\Psi_{a,b}(t)$ وجود دارد که آن ویژگی به بهترین وجه توسط این تابع تحلیل می‌شود. بنابراین، تبدیل موجک معادل با تحلیل سیگنال با فیلترهای میان‌گذری است که فرکانس میانی آن‌ها متغیر و وابسته به پارامتر a اما فاکتور

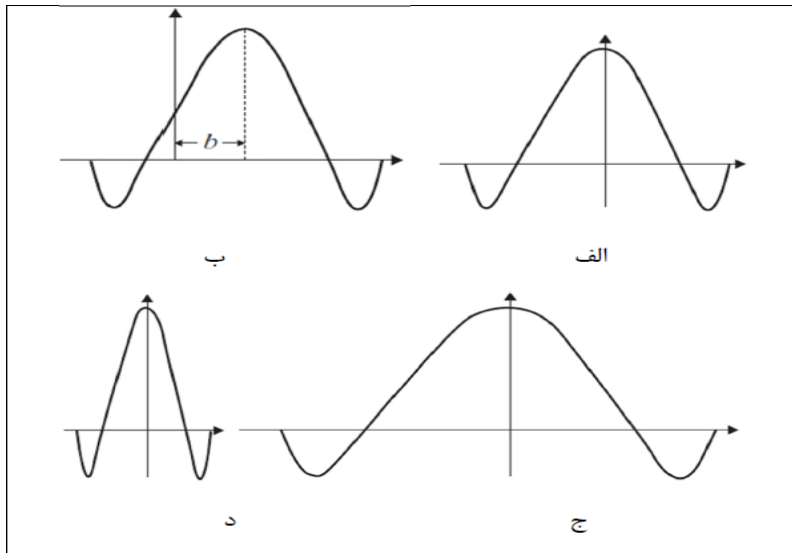
¹ Mother Wavelet

² Dilation

³ Translation

⁴ Natural Images

کیفیت^۱ آن‌ها ثابت است. توجه کنید که فاکتور کیفیت یک فیلتر میان‌گذر، نسبت فرکانس مرکزی به پهنای باند آن فیلتر تعریف می‌شود.



شکل ۳-۲: تأثیر مقیاس و جابجایی زمانی روی موجک مادر؛ (الف) موجک مادر $\Psi(t) = \Psi_{1,0}(t)$ با $a=1$ و $b=0$ (ب) حالت $a=1$ و $b \neq 1$ (ج) حالت $a=2$ و $b=0$ (د) حالت $a=1/2$ و $b=0$

۳-۴-۱ تبدیل موجک گسسته

تبدیل موجک تعریف‌شده در رابطه ۳-۱ سیگنال یک بعدی $X(t)$ را به تابعی دوبعدی مانند $X_w(a, b)$ نگاشت می‌کند و بنابراین موجب تولید تزیاید زیادی می‌شود. سیگنال اولیه را می‌توان از روی تبدیل موجک و به ازای مقادیر گسسته‌ای از پارامترهای a و b به دست آورد [۱۴]. پارامتر a را می‌توان با انتخاب $a = a_0^m$ (که $a_0 > 1$ و m عددی صحیح است) گسسته کرد. با افزایش مقدار a پهنای باند (یا همان درجه تفکیک فرکانسی) تابع پایه کاهش می‌یابد و بنابراین برای پوشش یک ناحیه‌ی فرکانسی مشخص، نیاز به گام‌های بیشتری (به نام سلول‌های درجه تفکیک) است. به طور مشابه، گسسته‌سازی پارامتر b نیز معادل با نمونه‌برداری زمانی است که فرکانس نمونه‌برداری وابسته به پهنای باند سیگنال نمونه‌برداری شده دارد که این پهنای باند نیز به نوبه خود به طور معکوس

¹ Quality Factor

متناسب با پارامتر a است. پارامتر b را می‌توان به صورت $b = nb_0 a_0^m$ انتخاب کرد. اگر $a_0 = 1$ و $b_0 = 1$ انتخاب شوند، برخی از درجات آزادی انتخاب تابع $\Psi(t)$ وجود دارند که موجب می‌شوند توابع $\Psi_{m,n}(t)$ تشکیل یک پایه‌ی یک-متعامد^۱ در فضای توابع مجذور-انتگرال‌پذیر دهند. این امر به این معنا است که هر تابع مجذور-انتگرال‌پذیر $X(t)$ را می‌توان به صورت یک ترکیب خطی از توابع پایه به صورت زیر توصیف کرد :

$$x(t) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} a_{m,n} \Psi_{m,n}(t) \quad (3-3)$$

که در آن، ضرایب $a_{m,n}$ به نام ضرایب تبدیل موجک تابع $X(t)$ نامیده شده و از روی رابطه ۱-۳ چنین محاسبه می‌شوند :

$$a_{m,n} = \int_{-\infty}^{\infty} x(t) \Psi_{m,n}(t) \quad (4-3)$$

جالب است توجه شود که با هر افزایش مقدار m مقدار a دو برابر می‌شود (یعنی دو برابر شدن عرض یا پهنای باند زمانی و نصف شدن عرض یا پهنای باند فرکانسی). این ویژگی متناظر با تحلیل سیگنال به کمک تجزیه‌ی اکتاو باندهای فرکانسی و نیز تبدیل موجک دوتایی^۲ است. با توجه به شکل ۱-۳ (که برای توضیح اصول عملکرد کدگذاری زیرباندی استفاده شد)، می‌توان نتیجه گرفت که تبدیل موجک در واقع نوعی کدگذاری زیرباندی است.

۳-۴-۲ نمایش چند درجه تفکیک

با استفاده از مفهوم تحلیل چند درجه تفکیک سیگنال^۳ می‌توان کاربرد تبدیل موجک در کدگذاری تصویر را بهتر درک کرد. فرض کنید تابعی مانند $\Phi(t)$ وجود دارد طوری که مجموع توابع $\Phi(t-n), n \in \mathbb{Z}$ یک - متعامد باشند. همچنین فرض کنید که تابع $\Phi(t)$ جواب یک معادله تفاضلی دومقیاسی به صورت زیر باشد :

¹ Orthonormal

² Dyadic Wavelet Transform

³ Multiresolution Signal Analysis

$$\Phi(t) = \sum_{n=-\infty}^{\infty} c_n \sqrt{2} \Phi(2t - n) \quad (5-3)$$

که در آن

$$c_n = \int_{-\infty}^{\infty} \Phi(t) \sqrt{2} \Phi(2t - n) dt \quad (6-3)$$

است. فرض کنید که $X(t)$ یک سیگنال مجذور-انتگرال پذیر باشد که بتوان آن را به صورت یک

ترکیب خطی از توابع $\Phi(t - n)$ توصیف کرد:

$$x(t) = \sum_{n=-\infty}^{\infty} c_n \Phi(t - n) \quad (7-3)$$

که در آن ضرایب بسط بوده و حاصل افکنش^۱ سیگنال $X(t)$ بر روی توابع $\Phi(t - n)$ می‌باشند. از آنجا که مقیاس^۲ یک تابع، درجه تفکیک آن را تغییر می‌دهد، می‌توان سیگنال $X(t)$ را بر حسب درجات تفکیک مختلف (که با مقیاس و فشردن تابع $\Phi(t)$ ایجاد می‌شوند) نمایش داد. بنابراین می‌توان سیگنال $X(t)$ را در هر درجه‌ی تفکیک m به صورت زیر نمایش داد:

اگر فضای تشکیل شده از توابع $2^{-m/2} \Phi(2^{-m}t - n)$ را V_m بنامیم، با توجه به رابطه ۵-۳ می‌توان

$$x_m(t) = 2^{-m/2} \sum_n c_n^m \Phi(2^{-m}t - n) \quad (8-3)$$

دید که تابع $\Phi(t)$ به گونه‌ای است که به ازای هر $j < i$ ، هر تابعی که متعلق به فضای V_i می‌باشد متعلق به فضای V_j نیز است و بدین ترتیب برای $j < i$ داریم $V_i \subset V_j$ بنابراین، فضاهای متناظر با مقیاس‌های متوالی تودرتو بوده و هر فضای V_m (با m در حال افزایش) را می‌توان به صورت فضایی که درجه‌ی تفکیک آن رو به کاهش است، نگاه کرد.

در نتیجه، فضای متناظر با درجه تفکیک درشت‌تر^۳ V_{j-1} را به دو زیر فضا می‌توان تجزیه کرد: یک فضا متناظر با درجه‌ی تفکیک ریزتر^۴ و فضای دیگر، فضای W_j که فضای مکمل متعامد فضای V_j بوده

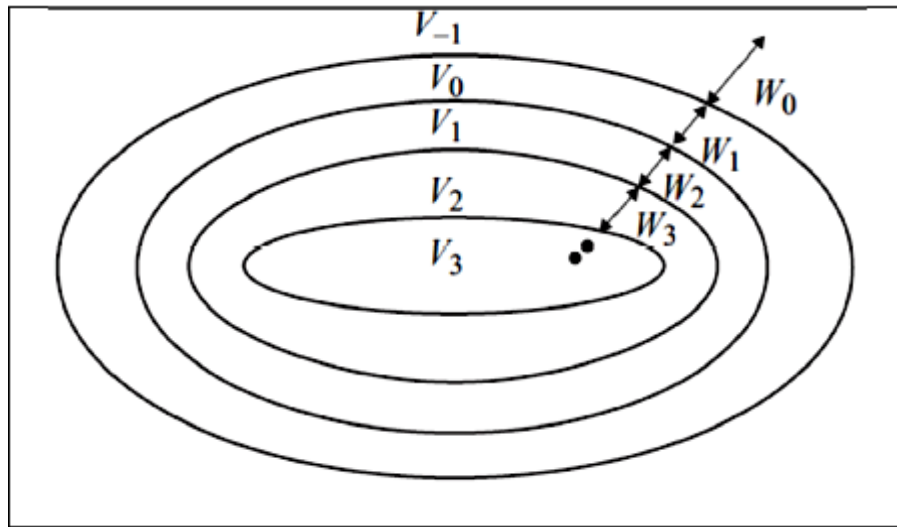
¹ Projection

² Dilation

³ Coarser Resolution

⁴ Finer Resolution

و به گونه‌ای است که $V_j + W_j = V_{j-1}$ گردد. در این رابطه (به دلیل متعامد بودن دو فضای W_j و V_j) داریم $W_j \perp V_j$. فضای W_j در واقع فضای ناشی از اختلافات بین درجات تفکیک نرم و درشت مذکور است و می‌توان از آن به صورت جزئیات لازم برای حرکت از فضای با درجه تفکیک کوچک‌تر، V_j به فضای با درجه تفکیک بزرگ‌تر، V_{j-1} یاد کرد. سلسله‌مراتب فضاها از دیدگاه درجه تفکیک در شکل ۳-۳ نشان داده شده است.



شکل ۳-۳: فضاها و چند درجه‌ی تفکیک

آقای مَلْت^۱ [۱۵] نشان داد که در حالت کلی، پایه‌ی متناظر با فضای W_j را می‌توان به صورت جابه‌جایی‌ها و مقیاس‌های یک تابع نمونه^۲ به نام موجک توصیف کرد. بنابراین، فضای W_m در واقع فضای تشکیل شده از توابع $\Psi_{m,n}(t) = 2^{-m/2}\Psi(2^{-m}t - n)$ می‌باشد. موجک $\Psi_t \in V_{-1}$ را می‌توان به صورت زیر از روی تابع $\Phi(t)$ به دست آورد:

$$\Psi_{(t)} = \sum_{n=-\infty}^{\infty} (-1)^n c_{1-n} \sqrt{2} \Phi(2t - n) \quad (9-3)$$

تابع $\Phi(t)$ را تابع مقیاس^۳ مربوط به نمایش چنددرجه‌ی تفکیک می‌نامند. بنابراین ضرایب تبدیل موجک رابطه ۳-۴ متناظر با حاصل افکنش سیگنال $X(t)$ بر روی فضای جزئیات متناظر با درجه‌ی

¹ Mallat

² Prototype

³ Scaling Function

تفکیک m ، یعنی W_m می‌باشند. بنابراین، تبدیل موجک اساساً یک سیگنال را به فضاها با درجات تفکیک مختلفی تجزیه می‌کند. در متون علمی، به این نوع تجزیه در حالت کلی، تجزیه چنددرجه‌ی تفکیک^۱ اطلاق می‌گردد.

در ادامه مثالی از نحوه‌ی محاسبه‌ی موجک هار^۲ به کمک تکنیک فوق، آورده می‌شود.

مثال (موجک هار)

تابع مقیاس متناظر با موجک هار همان تابع معروف مستطیلی است :

$$\Phi(t) = \begin{cases} 1 & 0 \leq t \leq 1 \\ 0 & \text{otherwise} \end{cases}$$

تابع فوق در رابطه ۵-۳ صدق کرده و ضرایب c_n مربوطه را می‌توان از رابطه ۶-۳ به صورت

$$c_n = \begin{cases} \frac{1}{\sqrt{2}} & n = 0, 1 \\ 0 & \text{otherwise} \end{cases}$$

محاسبه کرد. بنابراین با استفاده از رابطه ۹-۳ می‌توان موجک هار را به صورت زیر به دست آورد:

$$\Psi(t) = \Phi(2t) - \Phi(2t - 1)$$

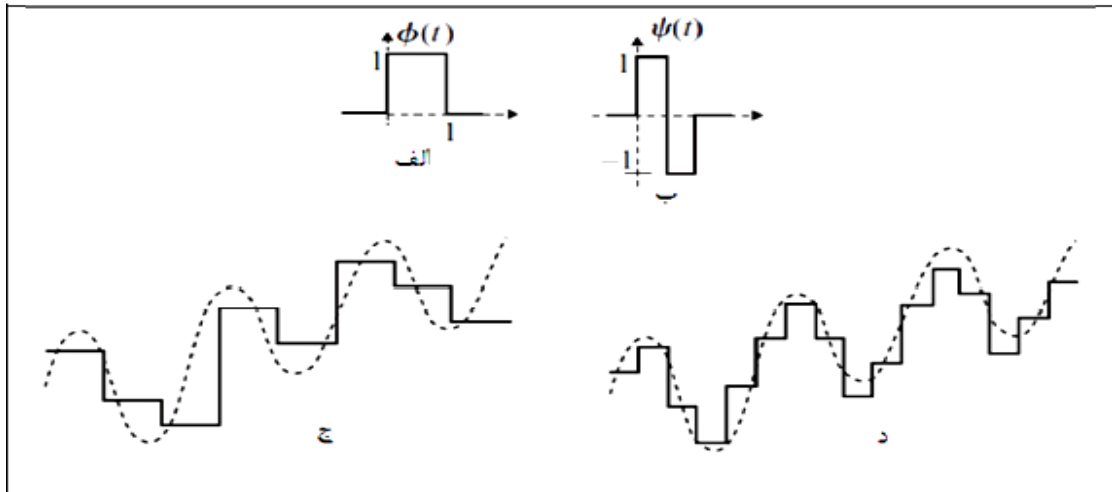
$$\Psi(t) = \begin{cases} 1 & 0 \leq t \leq 1/2 \\ -1 & 1/2 \leq t \leq 1 \end{cases}$$

تابع مقیاس $\Phi(t)$ (تابع مستطیلی) و موجک هار متناظر با آن به ترتیب شکل ۴-۳ الف و ب نشان

داده شده‌اند.

¹ Multiresolution Decomposition

² Haar Wavelet



شکل ۳-۴: (الف) تابع مقیاس هار . (ب) موجک هار . (ج) تقریب یک تابع پیوسته $X(t)$ در مقیاس $A_1x(t)$ (د) درشت تر $A_0x(t)$ و مقیاس نرم تر (یا بزرگ تر) $A_1x(t)$

از دیدگاه تقریب (سیگنال) می توان تجزیه ی چند درجه ی تفکیکی را چنین توضیح داد. فرض کنید سیگنال $x(t)$ در درجه تفکیک J توسط تابع $A_1x(t)$ و از طریق بسط دادن به کمک توابع پایه ی متعامد تقریب زده شود. فضای W_j فضای اختلافات بین مقیاس درشت تر V_{j-1} و مقیاس نرم تر V_j می باشد. سیگنال $D_jX(t) \in W_j$ نیز اختلاف بین دو تقریب از سیگنال $x(t)$ در دو درجه ی تفکیک J و $J-1$ است. در واقع داریم: $D_jx(t) = A_{j-1}x(t) - A_jx(t)$. بنابراین سیگنال $x(t)$ را می توان به دو قسمت تقسیم کرد:

$$x(t) = A_{-1}x(t) = A_0x(t) + D_0x(t)$$

شکل ۳-۴ ج و شکل ۳-۴ د نشان دهنده ی دو تقریب از یک تابع پیوسته در دو درجه ی تفکیک متوالی و به کمک تابع مقیاس مستطیلی می باشند. تقریب درشت تر $A_0x(t)$ در شکل ۳-۴ ج و تقریب انجام شده در درجه ی تفکیک بزرگ تر، یعنی $A_1x(t)$ نیز در شکل ۳-۴ د نشان داده شده اند.

در یک تابع هموار و با تغییرات آهسته ی $x(t)$ عمده ی تغییرات (که همان انرژی سیگنال می باشند) در جزء $A_0x(t)$ قرار گرفته و جزء $D_0x(t)$ تقریباً صفر می باشد. با تکرار فرآیند جداسازی فوق الذکر و انجام یک بخش بندی دیگر به صورت $A_0x(t) = A_1x(t) + D_1x(t)$ می توان تبدیل موجک سیگنال $x(t)$ را محاسبه کرده و تابع اولیه $x(t)$ را می توان برحسب موجک های آن به صورت زیر

بسط داد:

$$x(t) = D_0 x(t) + D_1 x(t) + D_2 x(t) + \dots + D_n x(t) + A_n x(t) \quad (10-3)$$

که در آن پارامتر n بیانگر تعداد سطوح تجزیه است. از آنجا که دامنه‌ی تغییرات^۱ علامت‌های جزئیات $D_j x(t)$ بسیار کمتر از دامنه‌ی تغییرات تابع اولیه‌ی $x(t)$ است، کدگذاری آن‌ها راحت تر از کدگذاری ضرایب بسط رابطه ۳-۳ است.

۳-۴-۳ تبدیل موجک و بانک فیلتری

برای اینکه فرآیند جداسازی مکرر گفته شده در بخش قبل در عمل قابل پیاده‌سازی باشد باید الگوریتم موثری برای محاسبه‌ی $D_j x(t)$ از روی ضرایب بسط اولیه‌ی سیگنال $x(t)$ در دست باشد. یکی از نتایج بخش‌بندی چنددرجه‌ی تفکیک فضا این است که تابع مقیاس $\Phi(t)$ دارای خاصیت خود تشابهی است. اگر $\Phi(t)$ و $\bar{\Phi}(t)$ به ترتیب توابع مقیاس تحلیل (یعنی مربوط به مسیر تجزیه) و ترکیب (یعنی مربوط به مسیر بازسازی) باشند و همچنین توابع $\Psi(t)$ و $\bar{\Psi}(t)$ به ترتیب موجک‌های تحلیل و ترکیب باشد، آن گاه از آن جا که $V_j \subset V_{j-1}$ می‌باشد، این توابع را می‌توان به صورت بازگشتی به صورت رابطه ۱۱-۳ تعریف کرد:

(۱۱-۳)

$$\Phi(t) = \sum_{n=-\infty}^{\infty} c_n \sqrt{2} \Phi(2t - n)$$

$$\bar{\Phi}(t) = \sum_{n=-\infty}^{\infty} \bar{c}_n \sqrt{2} \bar{\Phi}(2t - n)$$

$$\Psi(t) = \sum_{n=-\infty}^{\infty} d_n \sqrt{2} \Phi(2t - n)$$

$$\bar{\Psi}(t) = \sum_{n=-\infty}^{\infty} \bar{d}_n \sqrt{2} \Phi(2t - n)$$

این روابط بازگشتی راهی را برای محاسبه ضرایب تقریب $x(t)$ در درجه تفکیک j ، یعنی $A_j x(t)$ و

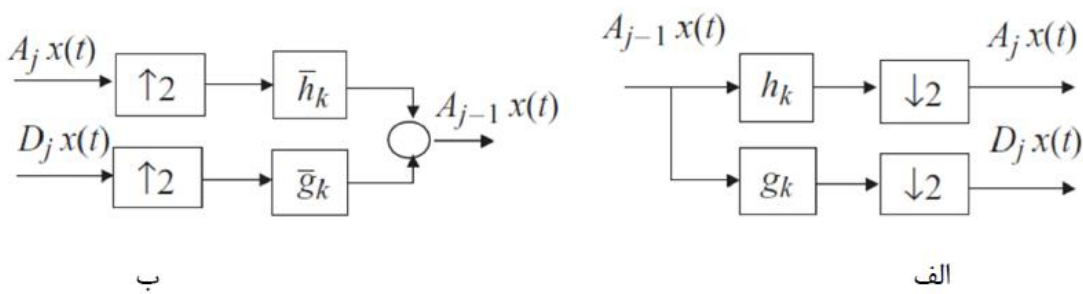
¹ Dynamic Range

ضرایب سیگنال جزئیات، یعنی $D_j x(t)$ از روی ضرایب تقریب سیگنال در مقیاس بالاتر، یعنی $A_{j-1} x(t)$ در اختیار قرار می‌دهند. به کمک برخی روابط و دستکاری‌های ریاضی می‌توان نشان داد که هر دو گروه ضرایب تقریب و جزئیات مربوط به یک درجه تفکیک ریزتر را می‌توان از طریق کانولوشن کردن ضرایب تقریب سیگنال در درجه‌ی تفکیک درشت‌تر با یک فیلتر خاص و سپس زیرنمونه‌برداری سیگنال حاصل شده با نرخ ۲ به دست آورد. برای این کار، در مورد ضرایب تقریب مربوط به درجه تفکیک ریزتر (یا پایین‌تر^۱) از یک فیلتر پایین‌گذر با ضرایب $h_k = c_{-k}$ و برای جزئیات نیز از یک فیلتر بالاگذر با ضرایب $g_k = d_{-k}$ استفاده می‌شود. به عکس، می‌توان سیگنال مربوط به درجه تفکیک بالاتر (یا درشت‌تر) را می‌توان از روی سیگنال جزئیات متناظر با آن درجه‌ی تفکیک و سیگنال تقریب مربوط به درجه‌ی تفکیک پایین‌تر بازیابی کرد. برای انجام این کار ابتدا ضرایب سیگنال‌های تقریب و جزئیات گفته‌شده را با نرخ ۲ بالانمونه‌برداری^۲ کرده، سپس به ترتیب فیلترهای ترکیب^۳ با ضرایب $\bar{g}_k = \bar{d}_{-k}$ و $\bar{h}_k = \bar{c}_{-k}$ کانولوشن کرده و در پایان حاصل به دست آمده را با هم جمع می‌کنیم. یک مرحله از هر کدام از فرآیندهای تفکیک (یا تجزیه) و ترکیب در شکل ۳-۵ نشان داده شده است. این شکل در واقع مشابه با نمایش بلوکی مربوط به کدگذاری زیرباندی می‌باشد. در فرآیند تجزیه، سیگنال ورودی به دو مؤلفه باند بالاگذر و باند پایین‌گذر تفکیک شده و در نتیجه درجه‌ی تفکیک فرکانسی با ضرایب ۲ افزایش می‌یابد. گرچه به دلیل استفاده از عمل زیرنمونه‌برداری (با نرخ ۲) درجه تفکیک زمانی با ضریب ۲ کاهش می‌یابد. در نتیجه، در مجموع هر مرحله از فرآیند تجزیه موجب بهتر / بیشتر شدن درجه‌ی تفکیک فرکانسی و کمتر شدن درجه‌ی تفکیک زمانی می‌گردد.

¹ Lower Resolution

² Up-Sampling

³ Synthesis

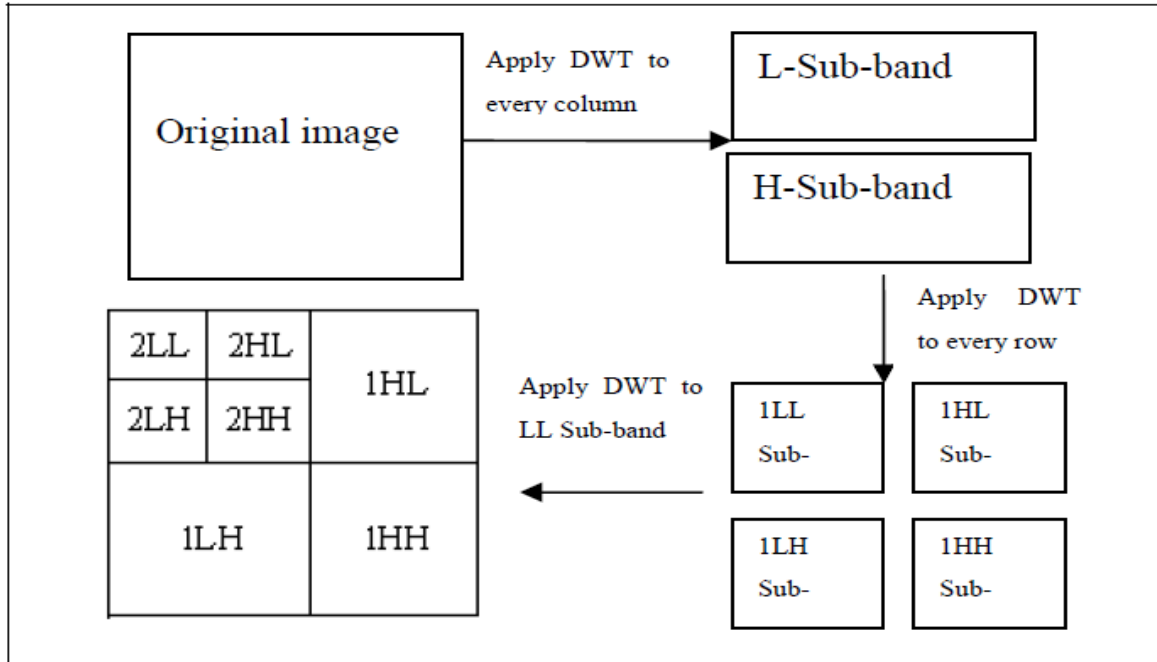


شکل ۳-۵: یک مرحله تبدیل موجک (الف) تحلیل (تجزیه) (ب) ترکیب

۳-۴-۴ اعمال تبدیل موجک بر روی تصویر [۱۶]

در واقع تصویر را می‌توان به عنوان یک سیگنال دو بعدی در نظر گرفت و با تعمیم ساختار فیلتر کردن دو بانندی نشان داده شده در شکل ۳-۵ در هر بعد از ابعاد سیگنال مورد بررسی، می‌توان به تبدیل موجک چندبعدی^۱ دست یافت. برای تجزیه‌ی یک تصویر (که سیگنالی دوبعدی محسوب می‌شود)، می‌توان فرآیند تجزیه‌ی یک بعدی را در هر کدام از ابعاد سطری و ستونی اعمال کرد. یک نمونه از چنین فرآیندی در شکل ۳-۶ نشان داده شده است. این شکل فرآیند تجزیه تصویر با استفاده از تبدیل موجک را نشان می‌دهد. در ابتدا فیلتر پایین‌گذر به هر یک از سطرهای تصویر اعمال می‌شود، که در نتیجه آن اجزای فرکانس پایین سطر به دست می‌آید ولی از آنجایی که فیلتر پایین‌گذر یک فیلتر نیم باند می‌باشد داده‌های خروجی شامل فرکانس‌هایی فقط در نیمه ابتدایی بازه فرکانسی اصلی می‌باشد. نمونه‌های خروجی زیرنمونه‌برداری با نرخ ۲ می‌شوند. بنابراین اطلاعات خروجی شامل نصف تعداد نمونه‌های اصلی می‌باشد. سپس فیلتر بالا‌گذر نیز به همان ترتیب به سطرها اعمال و اجزای فرکانس بالا بدست می‌آید.

¹ Multidimensional Wavelet Transform



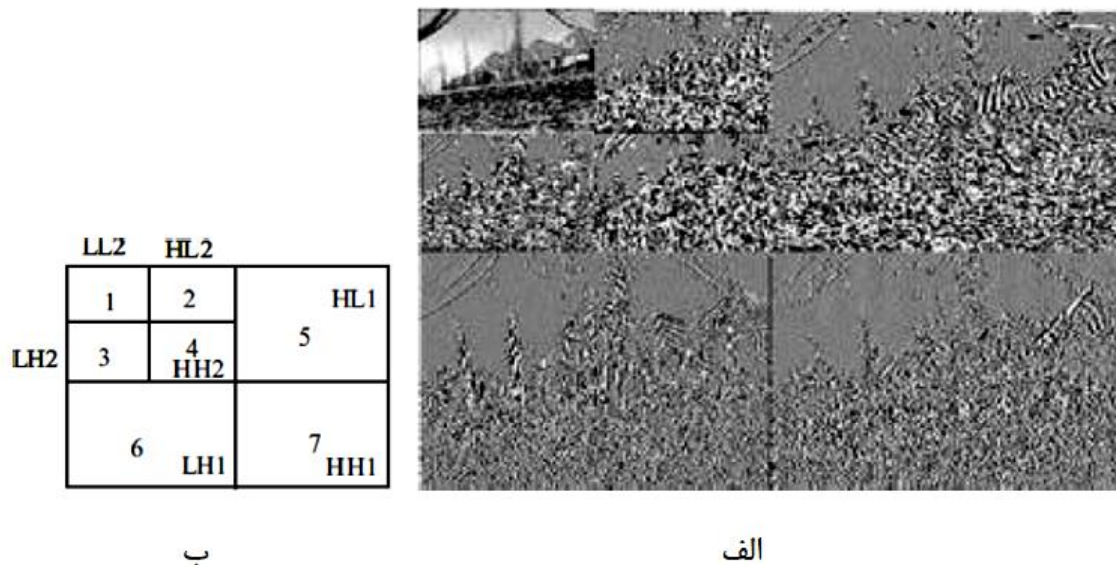
شکل ۳-۶: تبدیل موجک ۲ مرحله‌ای بر روی تصویر

در مرحله بعد فیلتر کردن به طریق گفته شده بر روی ستون‌های داده‌های خروجی مرحله قبل انجام می‌شود. این روند می‌تواند تا هر تعداد مرحله‌ای تکرار شود.

در شکل ۳-۶ نمادهای L و H به ترتیب بیانگر فیلترهای تجزیه‌ی پایین‌گذر و بالاگذر بوده که در آن‌ها از زیرنمونه‌برداری با نرخ ۲ نیز استفاده می‌شود. در اولین مرحله از این فرآیند تجزیه‌ی دوتایی^۱ سه زیرتصویر با محتویات فرکانس بالا تولید می‌شوند. برای مثال، زیرتصویر LH^1 عمدتاً دارای جزئیات فرکانس بالا (در راستای) عمودی و جزئیات فرکانس پایین (در راستای) افقی می‌باشد. این ویژگی‌ها در مورد زیرتصویر HL^1 برعکس می‌باشند. زیر تصویر HH^1 در هر دو راستای افقی و عمودی دارای جزئیات فرکانس بالا می‌باشد. جزئیات تصویر در درجه‌ی تفکیک پایین‌تر (یعنی مرحله‌ی دوم از فرآیند تجزیه‌ی نشان داده‌شده شکل ۳-۶ با نمادهای LH^2 ، HL^2 و HH^2 نمایش داده می‌شوند. باند LL^2 یک تصویر زیرنمونه‌برداری شده‌ی پایین‌گذر است که یک نسخه‌ی رونوشت برداری (کپی) شده از روی تصویر اصلی در ابعاد کوچکتر می‌باشد. حاصل اعمال فرآیند نشان داده‌شده که در شکل ۳-۶

¹ Dyadic Decomposition

روی یک نمونه تصویر شکل ۷-۳ الف نشان داده شده است (تصویر اصلی در این شکل نشان داده نشده است). در این فرآیند، هفت زیرتصویر تولید می‌شود که نام و جایگاه هر کدام در شکل ۷-۳ ب مشخص شده است.

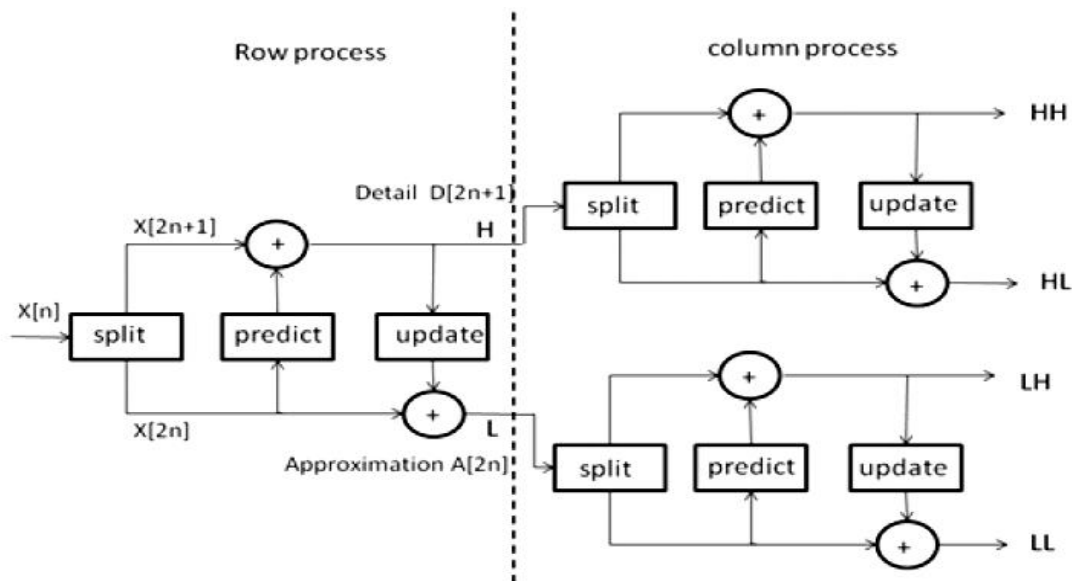


شکل ۷-۳: (الف) هفت زیرتصویر تولید شده توسط فرآیند نشان داده شده در شکل ۳-۶ (ب) چیدمان باندهای مختلف متناظر با فرآیند ۷-۳

۳-۴-۵ تبدیل موجک به روش طرح بالابر

روش جایگزین دیگر برای تبدیل موجک روش طرح بالابر می‌باشد، که ابتدا توسط ویم سویلدنز^۱ معرفی شد [۱۷]. همانطور که در بخش‌های قبل ذکر شد تبدیل موجک با استفاده از کانولوشن نیاز به حافظه و محاسبات زیادی دارد، بنابراین روش مناسبی برای پیاده‌سازی سخت‌افزاری در مواقعی که میزان حافظه و حجم محاسبات قابل انجام به شدت محدود است نمی‌باشد. در این مواقع طرح بالابر روش مناسبی به حساب می‌آید. در شکل ۳-۸ مراحل طرح بالابر دو بعدی که شامل پیش‌بینی و بروز رسانی است دیده می‌شود.

¹ Wim Sweldens



شکل ۳-۸: تبدیل به روش طرح بالابر دوبعدی

مزیت دیگر طرح بالابر این است که تبدیل معکوس آن نیز از همین روش پیروی می کند.

این روش فقط نیاز به جمع و شیفیت اعداد صحیح دارد که پیاده سازی آن را بسیار آسان می کند.

برای موجک 5/3 مرحله پیش بینی و به روزرسانی در (۳-۱۲) و (۳-۱۳) آورده شده اند [۱۸].

$$D[2n+1]=X[2n+1]-0.5(X[2n]+X[2n+2]) \quad (۱۲-۳)$$

$$A[2n]=X[2n]+0.25(D[2n-1]+D[2n+1]) \quad (۱۳-۳)$$

در معادلات بالا $X[2n+1]$ پیکسل های فرد و $X[2n]$ پیکسل های زوج که از یک سطر یا ستون

گرفته شده اند می باشند. هر سطر یا ستون به صورت زیر می باشند.

$$x[0] \ x[1] \ x[2] \ x[3] \ \dots$$

برای محاسبه تبدیل موجک نیاز است که پیکسل های سطرها یا ستون ها در یک زمان گرفته

شوند. در (۳-۱۲) و (۳-۱۳) نیاز است که $X[2n+1]$ قبل از $X[2n]$ محاسبه شود، بنابراین پیکسل های

فرد باید ابتدا پردازش شوند و سپس پیکسل های زوج. در هر مرحله تبدیل موجک ابتدا سطر و سپس

ستون های تصویر پردازش می شوند و در هر مرحله طول سیگنال نسبت به مرحله قبل نصف می شود.

۱ فصل چهارم: روش پیشنهادی برای

پیاده‌سازی تبدیل موجک گسسته دوبعدی

۱-۴ مقدمه

در این فصل، ابتدا نرم‌افزار استفاده‌شده در این پژوهش را معرفی می‌کنیم و سپس معماری پیشنهادی در این پایان‌نامه را برای پیاده‌سازی تبدیل موجک گسسته دو بعدی در FPGA توضیح می‌دهیم.

۲-۴ نرم‌افزار Xilinx ISE

نرم‌افزار Xilinx ISE یکی از بهترین ابزارها برای سنتز و آنالیز طراحی‌های توصیف‌شده توسط زبان‌های سخت‌افزاری VHDL و Verilog است. این نرم‌افزار متعلق به شرکت Xilinx بوده و به طور کلی می‌توان قابلیت‌های آن را به صورت زیر خلاصه نمود:

آنالیز زمانی طراحی‌ها - بررسی دیاگرام‌های RTL - شبیه‌سازی عملکرد طراحی در ازای ورودی‌های مختلف و ...

۳-۴ معماری پیشنهادی

در این پژوهش از روش طرح بالابر برای پیاده‌سازی تبدیل موجک گسسته دوبعدی استفاده شده است که نسبت به روش بانک‌های فیلتری روشی جدیدتر و دارای پیچیدگی‌ها و حجم محاسبات ریاضی کمتری است. موجکی که برای پیاده‌سازی استفاده شده است موجک ۵/۳ نام دارد که در فشرده‌سازی بدون اتلاف تصاویر کاربرد دارد. طبقات پیش‌بینی و به‌روزرسانی موجک ۵/۳ با روش طرح بالابر از روابط (۱۲-۳) و (۱۳-۳) بدست می‌آید که در فصل سوم بیان شد:

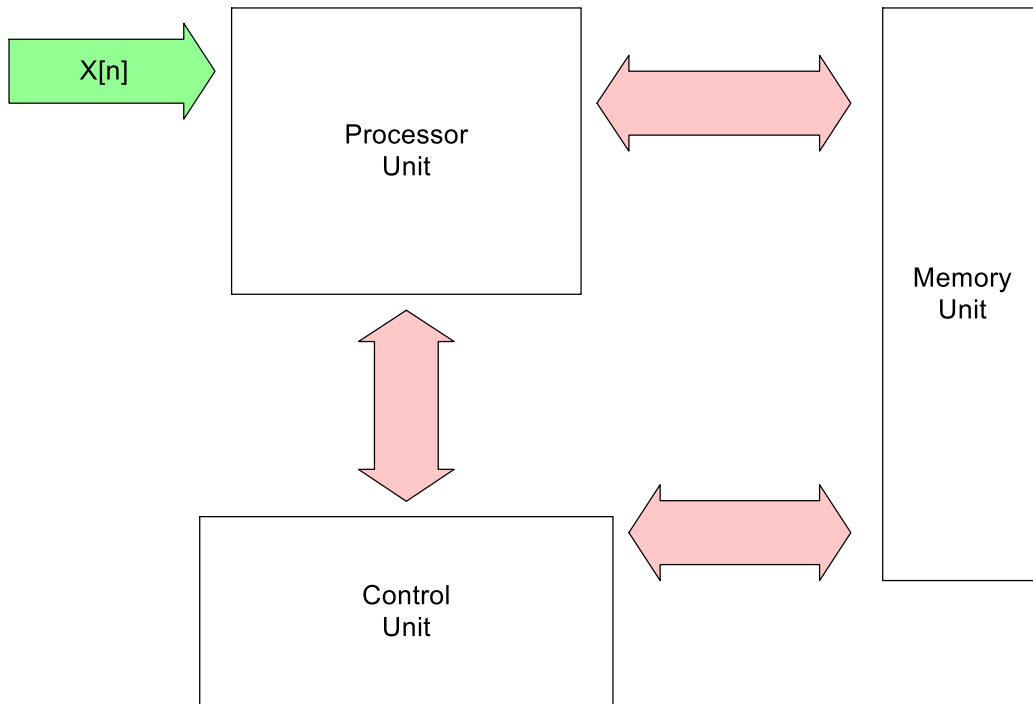
$$D[2n+1]=X[2n+1]-0.5(X[2n]+X[2n+2])$$

$$A[2n]=X[2n]+0.25(D[2n-1]+D[2n+1])$$

در روابط بالا $X[n]$ سیگنال ورودی، $D[2n+1]$ ضرایب جزییات تبدیل موجک گسسته و $A[2n]$ ضرایب تقریب تبدیل موجک گسسته می‌باشد.

۱-۳-۴ معماری ارائه شده

در شکل ۱-۴ نمایش بلوکی معماری پیشنهادی برای پیاده‌سازی تبدیل موجک گسسته دوبعدی نمایش داده شده است.



شکل ۱-۴: نمایش بلوکی معماری پیشنهادی

با توجه به این شکل معماری ارائه شده دارای سه بخش اصلی است:

۱- **واحد پردازنده:** که وظیفه انجام محاسبات مربوط به تبدیل موجک گسسته دوبعدی با روش طرح بالابر و موجک $5/3$ را بر عهده دارد. برای طراحی این واحد از روابط $3-12$ و $3-13$ استفاده شده است.

۲- **واحد کنترل:** که وظیفه کنترل حافظه برای ذخیره‌سازی ضرایب موجک و همچنین کنترل جریان داده در پردازنده برای انجام عملیات برش، به‌روز رسانی و پیش‌بینی را بر عهده دارد.

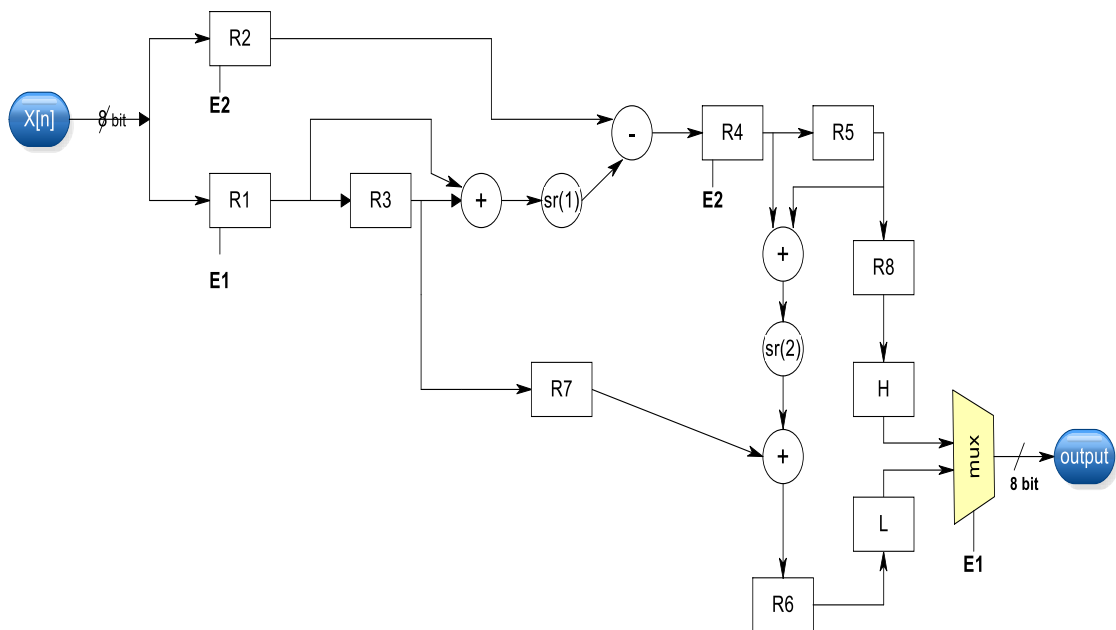
۳- **واحد حافظه خارجی:** که وظیفه ذخیره ضرایب تبدیل موجک گسسته و همچنین بازیابی آن‌ها برای انجام دوباره عملیات تبدیل موجک را بر عهده دارد.

در این معماری پیکسل‌های تصویر به ترتیب و به صورت داده‌های هشت بیتی وارد پردازنده می‌شوند و پس از اعمال عملیات تبدیل موجک بر روی آنها در حافظه خارجی ذخیره می‌شوند که خطوط آدرس حافظه برای ذخیره‌سازی این اطلاعات توسط واحد کنترل تولید می‌شوند. پس از آنکه همه پیکسل‌های تصویر وارد پردازنده شده و سپس نتایج آنها در حافظه ذخیره شد در مرحله بعد ضرایب تبدیل موجک ذخیره‌شده در حافظه به صورت ستونی بازیابی شده و وارد پردازنده می‌شوند تا به این ترتیب عملیات تبدیل موجک در راستای ستون‌های تصویر انجام می‌شود.

در ادامه اجزای این معماری به صورت کامل توضیح داده می‌شود.

۲-۳-۴ واحد پردازنده

در شکل ۲-۴ مدار مربوط به پردازنده تبدیل موجک ۵/۳ مشاهده می‌شود.



شکل ۲-۴: مدار مربوط به پردازنده تبدیل موجک ۵/۳

همان‌گونه که در شکل ۲-۴ مشاهده می‌شود پیکسل‌های تصویر ورودی به صورت یک بردار و در

اندازه‌های هشت بیتی به ترتیب وارد مدار پردازنده می‌شوند. در مدار پردازنده دو خط فعال ساز E1 و E2 قرار دارند که توسط واحد کنترل تولید می‌شوند و مقدار می‌گیرند. این دو خط فعال ساز بدین گونه عمل می‌کنند که در هر چرخه ساعت تنها یکی از آن‌ها فعال و دیگری غیرفعال است؛ پس زمانی که ثبات R1 فعال است ثبات R2 غیرفعال است و به همین دلیل پیکسل ورودی وارد R1 می‌شود. سپس در چرخه ساعت بعدی R1 غیرفعال می‌شود و R2 فعال می‌شود و پیکسل بعدی وارد R2 می‌شود. بدین ترتیب پیکسل‌های زوج تصویر از پیکسل‌های فرد جدا می‌شوند و عملیات برش روش طرح بالا بر تحقق می‌یابد. ثبات R2 پیکسل‌های فرد و ثبات R1 پیکسل‌های زوج را در خود قرار می‌دهد.

شرح کار مدار به این صورت است که ابتدا در چرخه ساعت اول یک پیکسل از تصویر وارد R1 می‌شود ($X[2n]$). سپس در دو چرخه ساعت بعد این پیکسل وارد R3 می‌شود و با پیکسل زوج بعدی که وارد R1 می‌شود ($X[2n+2]$) جمع می‌شود و یک واحد به سمت راست شیفت می‌یابد (0.5 $X[2n+1]$) و سپس با مقدار پیکسل موجود در R2 تفریق می‌شود ($X[2n+1]$) بدین ترتیب ضرایب جزییات تبدیل موجک گسسته تولید می‌شود ($0.5(X[2n]+X[2n+2])$). و بعد از چند چرخه ساعت وارد ثبات H می‌شود.

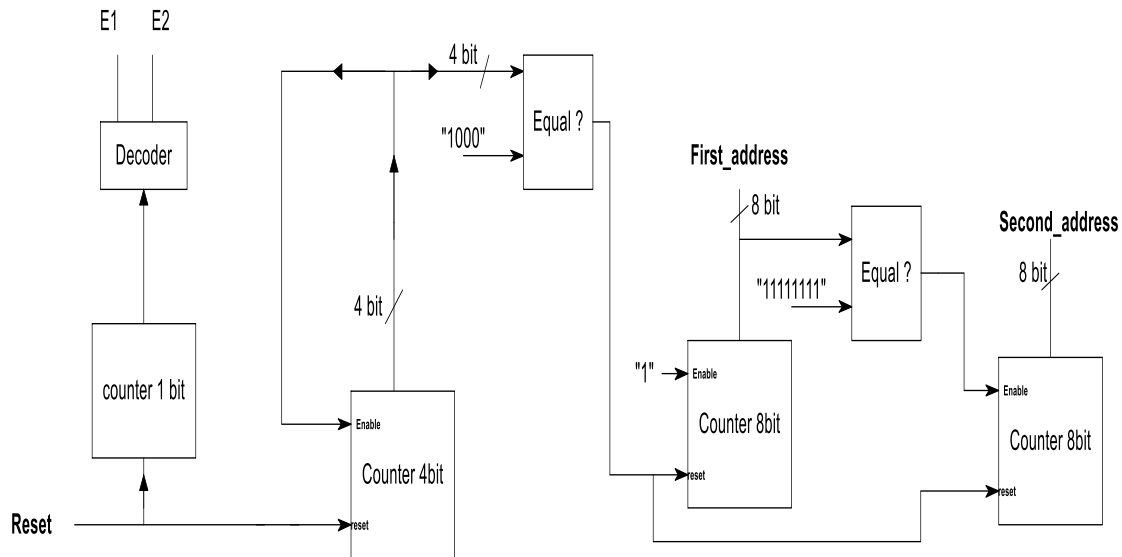
از سویی دیگر ضرایب جزییات تولیدشده در ثبات R5 ($D[2n-1]$) با ضرایب جزییات تولیدشده در دو چرخه ساعت بعد ($D[2n+1]$) جمع می‌شود و سپس دو واحد به سمت چپ شیفت می‌یابد ($0.25(D[2n-1]+D[2n+1])$) و با پیکسل زوج موجود در ثبات R7 جمع می‌شود تا ضرایب تقریب تبدیل موجک گسسته $5/3$ بدست آید.

ضریب تقریب در ثبات L و ضریب جزییات در ثبات H ذخیره می‌شود و توسط یک مالتی پلکسر ۲ به ۱ که سیگنال انتخاب آن E1 است به ترتیب وارد خروجی می‌شوند. ابتدا ضریب تقریب و سپس در چرخه ساعت بعدی ضرایب جزییات به خروجی پردازنده راه می‌یابند.

۳-۳-۴ واحد کنترل

در شکل ۳-۴ نمایش بلوکی واحد کنترل در معماری پیشنهادی برای پیاده‌سازی تبدیل موجک

گسسته با موجک ۵/۳ نشان داده شده است.



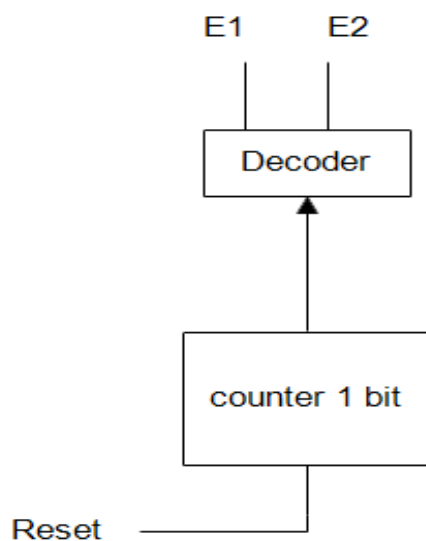
شکل ۳-۴: مدار مربوط به واحد کنترل معماری پیشنهادی

با توجه به شکل بالا مدار مربوط به واحد کنترل را می‌توان به دو بخش تقسیم کرد: بخش اول

مربوط به تولید خطوط فعال‌ساز برای واحد پردازنده است و بخش دوم مربوط به تولید گذرگاه آدرس

برای ذخیره‌سازی اطلاعات در حافظه خارجی است. در شکل ۴-۴ قسمت مربوط به پردازنده واحد

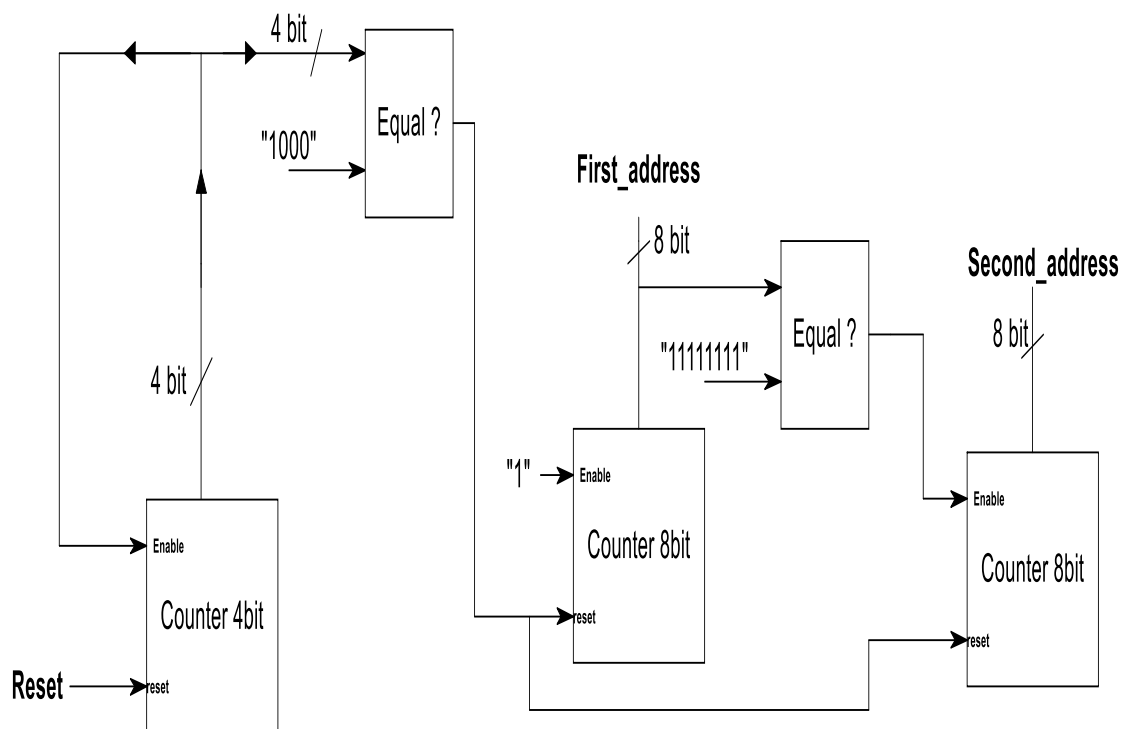
کنترل مشاهده می‌شود:



شکل ۴-۴: بخش مربوط به پردازنده واحد کنترل

با توجه به شکل ۴-۴ برای تولید خطوط فعال ساز E1 و E2 که در مدار پردازنده مورد استفاده قرار می‌گیرند از یک شمارنده یک بیتی و یک دیکدر ۱ به ۲ استفاده شده است. زمانی که سیگنال Reset فعال می‌شود شمارنده یک بیتی شروع به شمردن می‌کند و مرتب مقدار آن با هر تغییر لبه بالارونده ساعت به مقادیر صفر و یک منطقی تغییر می‌کند. زمانی که شمارنده مقدار "1" می‌پذیرد سیگنال E1 فعال می‌شود و در زمانی که شمارنده مقدار "0" می‌پذیرد سیگنال E2 فعال می‌شود و به همین صورت این روند ادامه می‌یابد.

در شکل ۴-۵ بخش مربوط به تولید گذرگاه آدرس واحد کنترل مشاهده می‌شود.



شکل ۴-۵: بخش مربوط به تولید گذرگاه آدرس واحد کنترل

با توجه به شکل بالا زمانی که Reset فعال می‌شود شمارنده چهار بیتی شروع به کار می‌کند و از صفر تا پانزده می‌شمرد و بعد متوقف می‌شود. در مدار پردازنده نه چرخه ساعت زمان می‌برد تا پیکسل‌های ورودی به خروجی برسند و خط لوله پردازنده پر شود. به همین دلیل این شمارنده چهار بیتی در مدار کنترل قرار داده شده است تا با ایجاد تأخیر، زمان مورد نیاز جهت پر شدن خط لوله فراهم شود. هنگامی که مقدار این شمارنده به هشت رسید ورودی Reset مربوط به دو شمارنده هشت بیتی فعال می‌شود و این دو شمارنده شروع به کار می‌کنند.

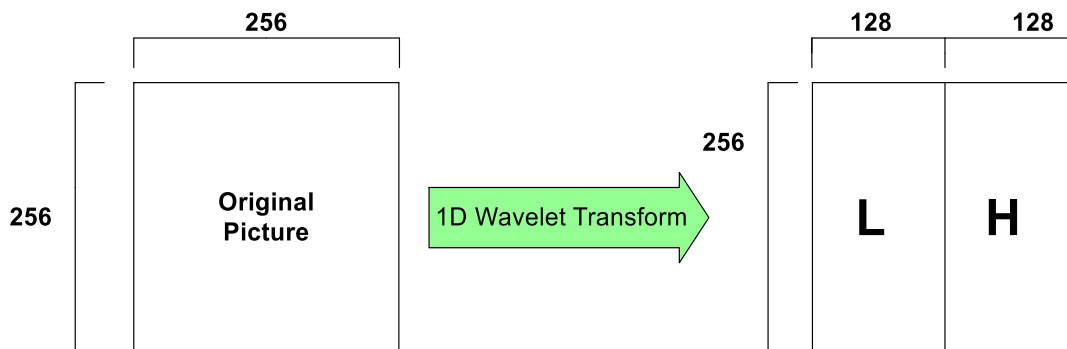
این دو شمارنده هشت بیتی برای تولید خطوط آدرس حافظه سطرها و ستون‌های تصویر ورودی که اندازه ابعاد آن $256 * 256$ است به کار گرفته شده‌اند. شمارنده هشت بیتی اول که سریعتر از شمارنده هشت بیتی دوم می‌شمرد آدرس حافظه مربوط به ستون‌های تصویر را تولید و شمارنده

هشت بیتی کندتر آدرس حافظه سطرهای تصویر را تولید می‌کند.

طرز کار این دو شمارنده به این صورت است که ابتدا شمارنده هشت بیتی اول از صفر تا ۲۵۵ می‌شمارد (آدرس ستون‌های یک سطر از تصویر تولید می‌شود) و هنگامی که مقدار First_address برابر با ۲۵۵ شد یعنی زمانی که تولید آدرس درایه‌های یک سطر از تصویر تمام شده است آنگاه شمارنده هشت بیتی دوم فعال شده و یک واحد به Second_address اضافه می‌شود (تولید آدرس یک سطر تمام شده و به سطر بعدی می‌رود). سپس شمارنده دوم غیرفعال شده و شمارنده اول بار دیگر از صفر تا ۲۵۵ می‌شمارد و بار دیگر Second_address یک واحد به آن اضافه می‌شود و این روند تا زمانی که تولید آدرس برای تمام سطرهای تصویر انجام شود ادامه می‌یابد.

۴-۳-۴ واحد حافظه

حافظه خارجی برای ذخیره‌سازی ضرایب تبدیل موجک به کار گرفته شده است. این ضرایب برای ذخیره شدن در حافظه نیاز به آدرس حافظه دارند که این آدرس‌ها در واحد کنترل تولید می‌شوند. اندازه حافظه نیز برابر اندازه تصویر ورودی است یعنی $256 * 256$ بایت. هنگامی که از یک تصویر در راستای سطرها تبدیل موجک گرفته می‌شود ابعاد تصویر خروجی به صورت شکل ۴-۳ در می‌آید.



شکل ۴-۳: اعمال تبدیل موجک یک بعدی بر تصویر ورودی

همانطور که در شکل ۴-۶ مشاهده می‌شود بعد از اعمال تبدیل موجک بر روی سطرهای تصویر (تبدیل موجک یک بعدی) تصویر خروجی به دو قسمت ضرایب تقریب (L) و ضرایب جزئیات (H) تبدیل می‌شود که اندازه هر کدام $128 * 256$ است؛ بنابراین ذخیره ضرایب تقریب و جزئیات موجک در حافظه خارجی نیز باید همین صورت انجام می‌شود؛ از آنجا که خروجی اطلاعات در واحد پردازنده در یک چرخه ساعت، ضریب تقریب و در چرخه بعدی، ضریب جزئیات است با استفاده از خطوط آدرس ضرایب تقریب به صورت مرتب در یک قسمت از حافظه و ضرایب جزئیات نیز در قسمت دیگر حافظه به ترتیب ذخیره می‌شوند.

پس از اعمال تبدیل موجک بر روی سطرهای تصویر، در مرحله بعد باید از ستون‌های تصویر خروجی تبدیل موجک گرفته شود تا ضرایب تبدیل موجک گسسته دو بعدی به دست آید؛ بنابراین پس از آنکه ضرایب موجک در حافظه خارجی ذخیره شد، بار دیگر همه پیکسل‌های تصویر خروجی به صورت ستونی از حافظه خوانده می‌شوند و وارد پردازنده می‌شوند تا دوباره مورد پردازش قرار گیرند تا بدین ترتیب ضرایب تبدیل موجک گسسته دو بعدی حاصل شوند.

فصل پنجم : نتایج حاصل از شبیه‌سازی و

مقایسه با کارهای انجام شده

۱-۵ مقدمه

در این فصل به بیان نتایج حاصل از کدنویسی و شبیه‌سازی معماری پیشنهادی با نرم‌افزار Xilinx ISE می‌پردازیم و تعداد منابع سخت‌افزاری که در پیاده‌سازی این معماری استفاده شده و همچنین حداکثر فرکانس کاری را بیان و خروجی برنامه را با تصاویر ورودی مختلف مشاهده و ارزیابی می‌کنیم و در نهایت نتایج بدست آمده را با برنامه تبدیل موجک گسسته نوشته شده در محیط متلب و همچنین با نتایج مقالات دیگر مورد مقایسه و ارزیابی قرار می‌دهیم.

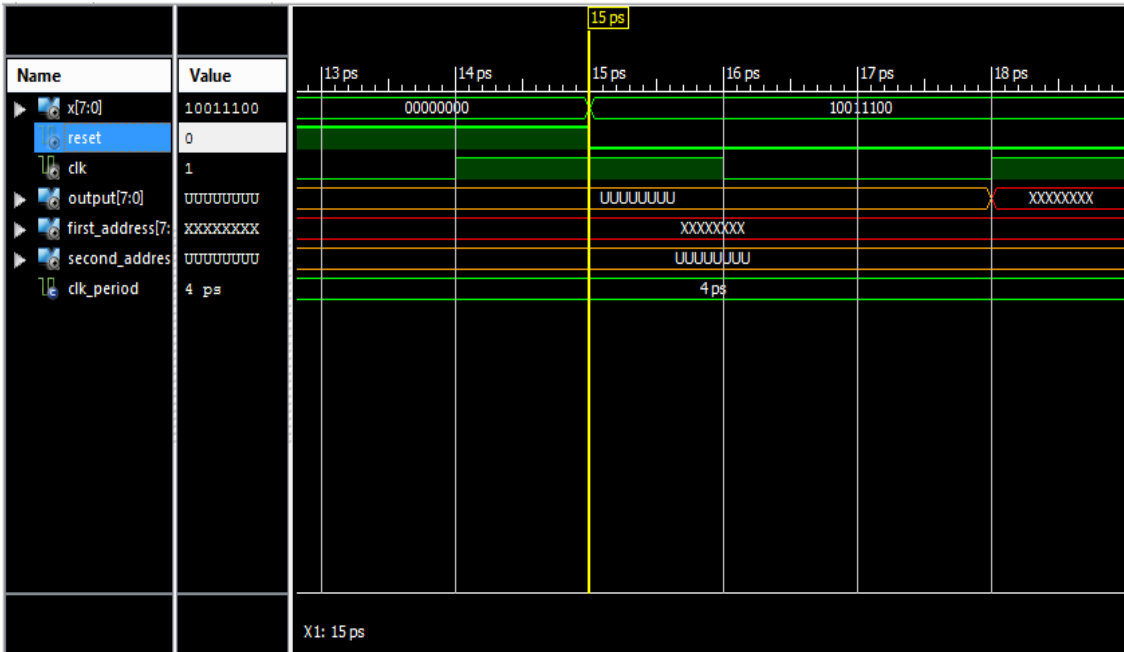
۲-۵ نتایج

معماری پیشنهادی به زبان VHDL نوشته شده و با نرم‌افزار Xilinx ISE شبیه‌سازی شده است. تراشه FPGA به کار گرفته شده در نرم‌افزار Xilinx ISE نیز از خانواده Spartan 3 و به نام XC3S500E می‌باشد.

تصاویری که به عنوان ورودی برنامه FPGA استفاده می‌شوند دارای ابعاد $256 * 256$ پیکسل هستند که تصاویر داخل نرم‌افزار متلب نیز دارای همین ابعاد هستند. از آنجا که همه پیکسل‌ها دارای مقادیر بین صفر تا ۲۵۵ می‌باشند و با هشت بیت قابل نمایش هستند؛ پیکسل‌ها به صورت کلمات هشت بیتی به عنوان ورودی، وارد پردازنده می‌شوند.

در زبان برنامه‌نویسی VHDL آرایه‌های دوبعدی و یا ماتریس قابل تعریف یا شناسایی نیستند؛ بنابراین تصاویر که آرایه‌های دو بعدی هستند باید به صورت بردار یا آرایه یک بعدی تبدیل شوند. جهت این کار در محیط نرم‌افزار متلب پیکسل‌های تصویر ابتدا به اعداد باینری تبدیل شده و سپس به صورت بردار در یک فایل متنی ذخیره می‌شوند تا به عنوان ورودی، وارد مدار شوند.

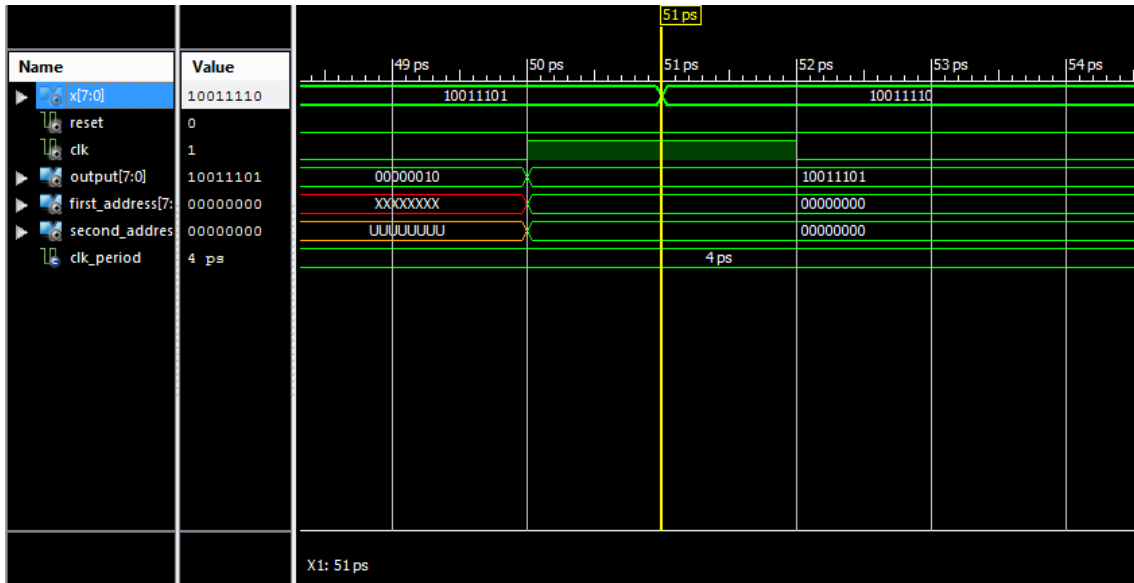
بعد از پیاده‌سازی و اجرای برنامه شکل موج ورودی و خروجی مدار به صورت شکل ۱-۵ است:



شکل ۱-۵: شکل موج ورودی و خروجی معماری پیشنهادی در زمانی که مدار شروع به کار می کند

همان گونه که در شکل ۱-۵ مشاهده می شود مدت زمان یک دوره چرخه ساعت (clk_period) ps ۴ در نظر گرفته شده است. زمانی که مقدار reset یک منطقی به خود پذیرفت و در چرخه ساعت بعدی مقدار آن صفر شد معماری شروع به کار کرده و پیکسل ورودی (X[7:0]) وارد مدار می شود. از آنجا که هنوز خط لوله پردازنده پر نشده است و ورودی به خروجی نرسیده است خروجی در این زمان صفر است.

در شکل ۲-۵ مشاهده می شود که بعد از نه چرخه ساعت از فعال شدن مدار، خط لوله پردازنده پر شده و خروجی نمایان می شود. سپس در چرخه ساعت بعدی شمارنده های first_address و second_address نیز فعال می شوند تا پیکسل های خروجی در حافظه ذخیره شوند.



شکل ۲-۵: شکل موج ورودی و خروجی مدار پیشنهادی پس از گذشت نه چرخه ساعت از فعال شدن Reset

۱-۲-۵ تصاویر خروجی

بعد از اجرای برنامه مقادیر خروجی که در واقع ضرایب تبدیل موجک یک بعدی و دو بعدی می‌باشند در یک فایل متنی ذخیره می‌شوند. با استفاده از نرم‌افزار متلب می‌توان این ضرایب را نمایش داد و تصاویر خروجی را مشاهده نمود.

تصویر Camera man : خروجی برنامه VHDL به ازای تصویر Camera man به صورت شکل

۳-۵ است:



الف



(ج)

(ب)

شکل ۳-۵: الف) تصویر ورودی ب) تبدیل موجک یک بعدی تصویر ج) تبدیل موجک دوبعدی تصویر

در شکل ۳-۵ الف تصویر ورودی مشاهده می شود که ابعاد آن $256 * 256$ است. در شکل ۳-۵ ب تبدیل موجک یک بعدی تصویر مشاهده می شود که دارای دو قسمت تقریب (L) و جزئیات (H) است که ابعاد آن هر کدام $128 * 256$ می باشد. در شکل ۳-۵ ج نیز تبدیل موجک دو بعدی تصویر مشاهده می شود که دارای چهار قسمت می باشد که ابعاد هر کدام از آنها $128 * 128$ است.

در شکل ۴-۵ تبدیل موجک دو بعدی دو سطحی تصویر Camera man مشاهده می شود که از

قسمت LL دوباره تبدیل موجک دو بعدی گرفته شده است.



شکل ۴-۵: تبدیل موجک دو بعدی دو سطحی تصویر

در شکل ۵-۵ تصویر خروجی متلب با تصویر خروجی برنامه VHDL مقایسه شده است.



(الف)

(ب)

شکل ۵-۵: (الف) تصویر خروجی برنامه VHDL (ب) تصویر خروجی متلب

همانطور که در شکل ۵-۵ مشاهده می شود تصویر خروجی برنامه VHDL اندکی با تصویر خروجی متلب اختلاف دارد که علت این اختلاف را می توان این گونه توجیه کرد که محاسبات در برنامه VHDL به روش ممیز ثابت انجام شده است در حالی که در برنامه متلب محدودیتی در انجام

محاسبات لحاظ نشده است.

تصویر سکه: خروجی برنامه VHDL به ازای تصویر سکه به صورت شکل‌های زیر است:



شکل ۵-۶: تصویر ورودی



(ب)

(الف)

شکل ۵-۷: (الف) تبدیل موجک یک بعدی تصویر (ب) تبدیل موجک دو بعدی تصویر

در شکل ۵-۸ تبدیل موجک دو بعدی دو سطحی تصویر سکه مشاهده می‌شود که از قسمت LL

دوباره تبدیل موجک دو بعدی گرفته شده است.



شکل ۵-۸: تبدیل موجک دو بعدی دو سطحی تصویر

در شکل ۵-۹ تصویر خروجی متلب با تصویر خروجی برنامه VHDL مقایسه شده است.



(الف)

(ب)

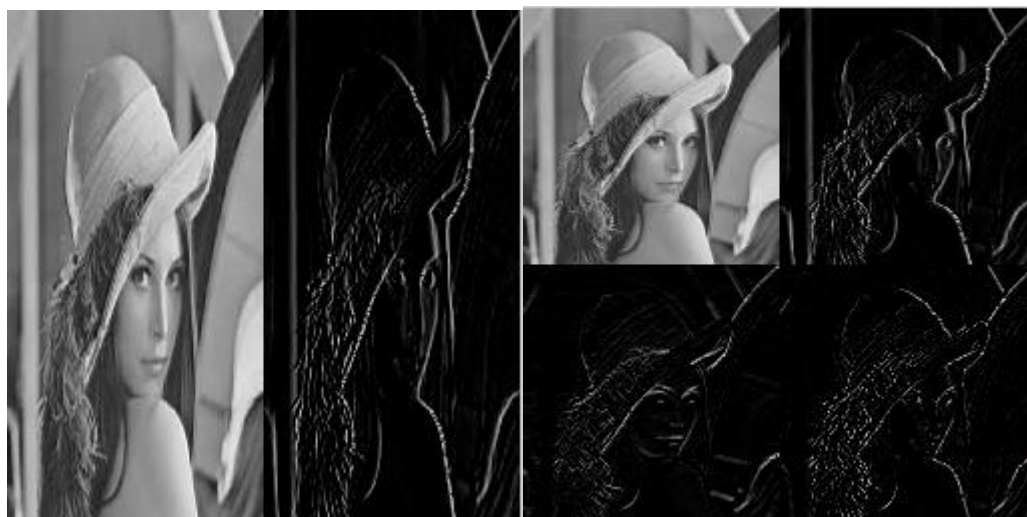
شکل ۵-۹: (الف) تصویر خروجی برنامه VHDL (ب) تصویر خروجی متلب

تصویر Lenna: در شکل زیر تصویر Lenna مشاهده می‌شود که به عنوان تصویر ورودی وارد برنامه می‌شود.



شکل ۵-۱۰: تصویر Lenna

در شکل ۵-۱۱ الف تبدیل موجک یک بعدی و در شکل ۵-۱۱ ب تبدیل موجک دو بعدی تصویر Lenna مشاهده می‌شود.



(الف)

(ب)

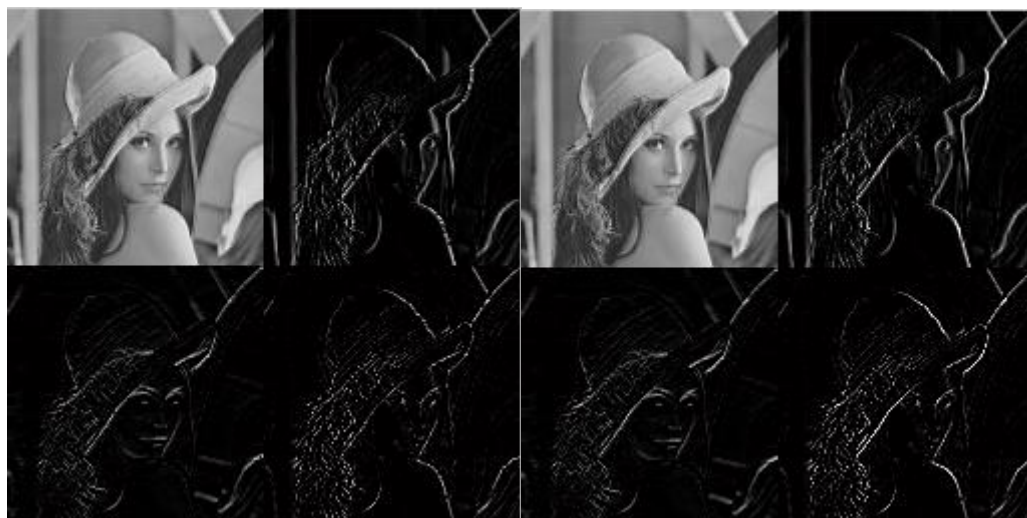
شکل ۵-۱۱: الف) تبدیل موجک یک بعدی تصویر ب) تبدیل موجک دو بعدی تصویر

در شکل ۱۲-۵ تبدیل موجک دو بعدی دو سطحی تصویر Lenna مشاهده می‌شود که از قسمت LL دوباره تبدیل موجک دو بعدی گرفته شده است.



شکل ۱۲-۵: تبدیل موجک دو بعدی دو سطحی تصویر

در شکل ۱۳-۵ تصویر خروجی متلب با تصویر خروجی برنامه VHDL مقایسه شده است.



(الف)

(ب)

شکل ۱۳-۵: الف) تصویر خروجی برنامه VHDL (ب) تصویر خروجی متلب

۲-۲-۵ فرکانس کاری و منابع استفاده شده

در جدول ۱-۵ تعداد منابع سخت‌افزاری استفاده‌شده در پیاده‌سازی معماری پیشنهادی و همچنین فرکانس کاری مشاهده می‌شود. همان‌گونه که در این جدول مشاهده می‌شود در ستون اول تعداد منابع استفاده‌شده از هر کدام از قطعات تراشه FPGA در معماری پیشنهادی بیان شده است. در ستون دوم تعداد کل منابع قابل دسترس از هر کدام از قطعات در تراشه Spartan 3 مشاهده می‌شود و در نهایت ستون سوم تعیین می‌کند که چند درصد از تعداد کل منابع قابل دسترس از هر قطعه در معماری پیشنهادی استفاده شده است.

جدول ۱-۵: تعداد منابع سخت‌افزاری استفاده شده در معماری پیشنهادی

	استفاده شده	در دسترس	%
Number of slices	۷۸	۴۶۵۶	%۱
Number of Slice Flip Flops	۱۱۹	۹۳۱۲	%۱
Number of 4input LUTs	۷۲	۹۳۱۲	%۰
Number of bonded IOBs	۳۴	۶۶	%۵۱
Number of GCLKs	۱	۲۴	%۴

Maximum Frequency	۱۷۲.۷۷۶ MHZ
--------------------------	-------------

۳-۵ مقایسه نتایج برنامه با سایر مقالات

در این بخش نتایج حاصل از معماری پیشنهادی با سایر مقالات مرتبط با این پژوهش مورد مقایسه و ارزیابی قرار می‌گیرد.

در جدول ۲-۵ تعداد منابع استفاده شده در معماری پیشنهادی با نتایج مقاله‌های [۹] و [۱۰] مقایسه شده است. با مقایسه نتایج مشاهده می‌شود که در معماری پیشنهادی از منابع کمتری نسبت به سایر مقالات استفاده شده است و به همین دلیل بهینه‌تر می‌باشد.

جدول ۲-۵: مقایسه نتایج معماری پیشنهادی با نتایج سایر مقالات

	مقاله [۹]	مقاله [۱۰]	روش پیشنهادی
Number of slices	۱۲۹۹	۲۷۲	۷۸
Number of Slice Flip Flops	۷۶۷	۴۹۷	۱۱۹
Number of 4input LUTs	-	۳۴۴	۷۲
Number of bonded IOBs	-	۹۱	۳۴
Number of GCLKs	-	۱	۱

در جدول ۳-۵ حداکثر فرکانس کاری معماری پیشنهادی با حداکثر فرکانس کاری سایر مقالات

مقایسه می‌شود.

جدول ۳-۵: مقایسه حداکثر فرکانس معماری پیشنهادی با فرکانس سایر مقالات

	مقاله [۹]	مقاله [۱۰]	معماری پیشنهادی
Maximum Frequency	۶۲.۷۹۷ MHZ	۲۰۵.۴۲۳ MHZ	۱۷۲.۷۷۶MHZ

فصل ششم: نتیجه‌گیری و پیشنهادها

۱-۶ مروری بر پژوهش انجام شده

در این پایان‌نامه یک معماری جدید برای پیاده‌سازی تبدیل موجک دوبعدی گسسته با روش طرح بالابر ارائه گردید که از موجک $5/3$ برای پیاده‌سازی استفاده شده است. معماری پیشنهادی شامل یک واحد پردازنده برای انجام محاسبات مربوط به تبدیل موجک مبتنی بر طرح بالابر و یک حافظه خارجی به اندازه ابعاد تصویر و یک واحد کنترل برای کنترل جریان داده در پردازنده و همچنین تولید خطوط آدرس برای ذخیره‌سازی ضرایب موجک در حافظه می‌باشد.

تصویر ورودی به ابعاد $256*256$ به صورت یک آرایه یک بعدی و در قالب اعداد باینری وارد مدار برنامه می‌شود. از نرم‌افزار Xilinx Ise برای کدنویسی و شبیه‌سازی معماری استفاده شده است و شکل موج خروجی برنامه و تعداد منابع سخت‌افزاری مورد استفاده و همچنین فرکانس کاری برنامه مشاهده و نتایج آن در فصل پنجم بیان شد. برنامه با تصاویر ورودی مختلف ارزیابی شد و تصویر خروجی برنامه با تصویر خروجی نرم‌افزار متلب مقایسه شد.

نتایج خروجی برنامه با نتایج سایر مقالات نیز مقایسه شد و نتایج این مقایسه نشان داد که معماری پیشنهادی از منابع سخت‌افزاری کمتری نسبت به این مقالات استفاده کرده است و بهینه‌تر است.

۲-۶ پیشنهادها

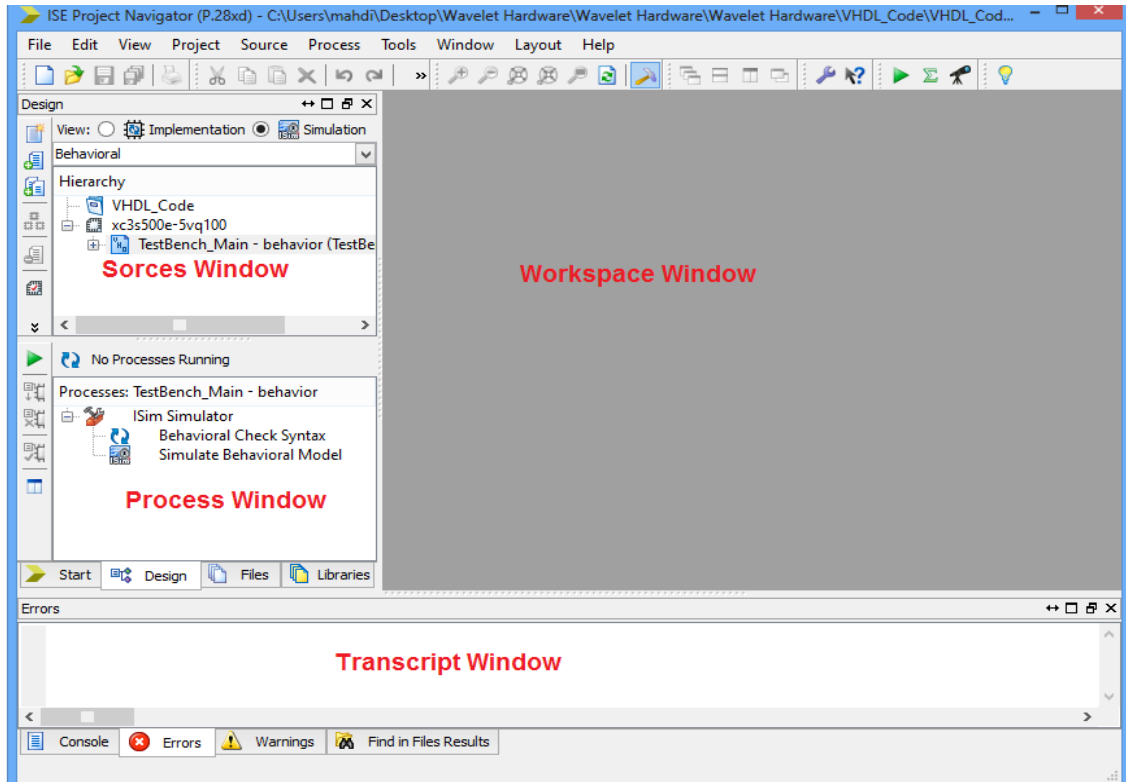
در این بخش چند پیشنهاد جهت ادامه و بهبود کار انجام شده ارائه می‌گردد:

- ابعاد تصویر ورودی برنامه $256*256$ است. می‌توان این معماری را با تصاویر ورودی با ابعاد بزرگتر مثلاً $512*512$ ارتقا داد.
- این معماری از تصویر ورودی تبدیل موجک دوبعدی می‌گیرد. می‌توان معماری ارائه داد که معکوس تبدیل موجک را پیاده‌سازی کند؛ یعنی ضرایب تبدیل موجک را به عنوان ورودی دریافت و عملیات معکوس تبدیل موجک را انجام دهد و تصویر اولیه را ایجاد کند.

پیوست‌ها

۱- آشنایی با نرم افزار

در شکل ۱ تصویری از قاب طراحی نرم افزار را نشان می دهد.



شکل ۱: قاب طراحی نرم افزار ISE

همانطور که در شکل ۱ مشاهده می شود قاب طراحی این نرم افزار شامل چهار پنجره است:

پنجره منابع

نام پروژه به همراه فایل های متعلق به آن و همچنین نوع FPGA انتخاب شده برای سنتز طراحی

مورد نظر در این پنجره نمایش داده می شود.

پنجره فرآیندها

تمام فرایندهای موجودی را که برای یک فایل انتخابی می توانند اجرا شود را نشان می دهد.

پنجره رونوشت

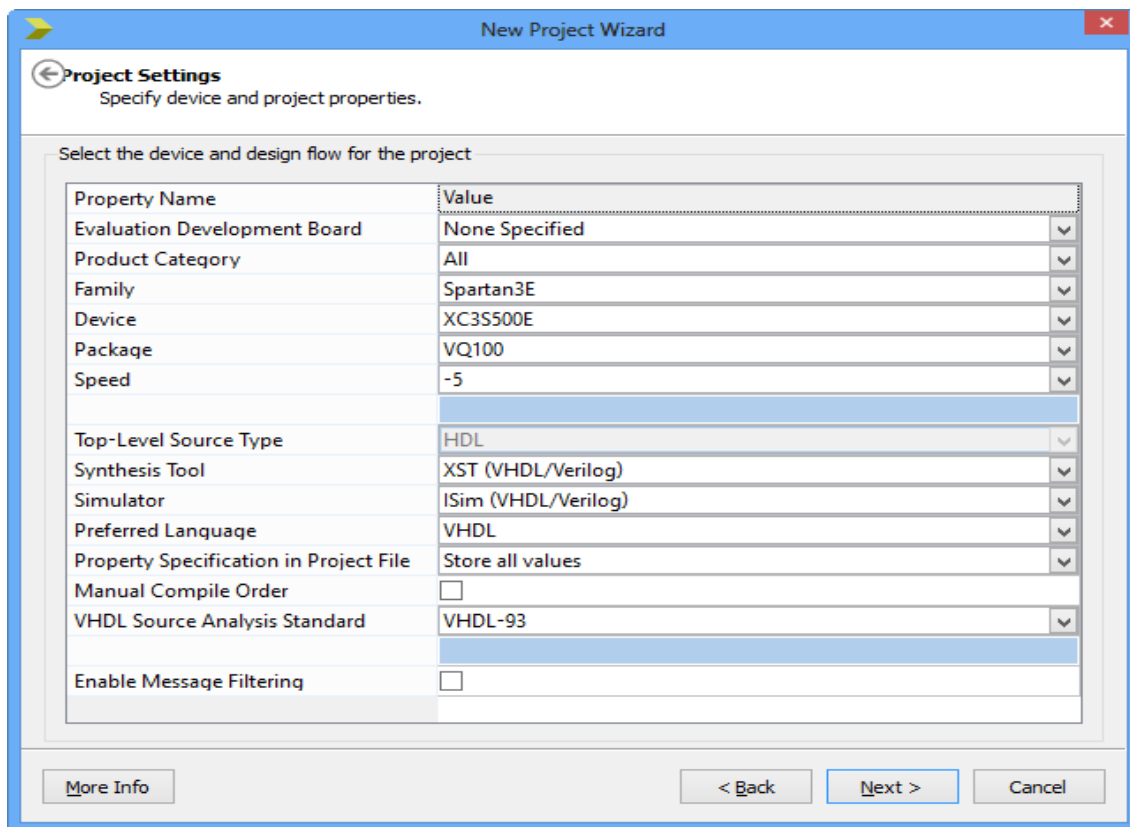
پیام‌های مورد نیاز برای طراح از قبیل خطاهای موجود در طراحی که مانع تکمیل فرایند سنتز هستند، در این پنجره نشان داده می‌شوند و بدین ترتیب طراح می‌تواند با برطرف نمودن آن‌ها، سنتز مدار خود را با موفقیت به اتمام برساند.

پنجره فضای کار

در حقیقت این بخش محیط کار طراح بوده و توصیف مدار یا طراحی شماتیک مورد او در این قسمت انجام می‌گیرد.

۲- ایجاد یک پروژه جدید

برای شروع کار لازم است که یک پروژه جدید ایجاد کنیم. برای این کار از قسمت File گزینه Project New را انتخاب می‌کنیم. در پنجره ایجاد شده نام پروژه و محل ذخیره‌سازی پروژه را مشخص می‌کنیم و سپس در پنجره بعدی مانند شکل ۲ مشخصات آی سی FPGA و نوع زبان برنامه‌نویسی را تعیین می‌کنیم.



شکل ۲: تعیین مشخصات آی سی FPGA برای ایجاد یک پروژه در نرم افزار ISE

و در نهایت با کلیک بر روی دکمه Next و سپس Finish پروژه جدید ساخته می شود.

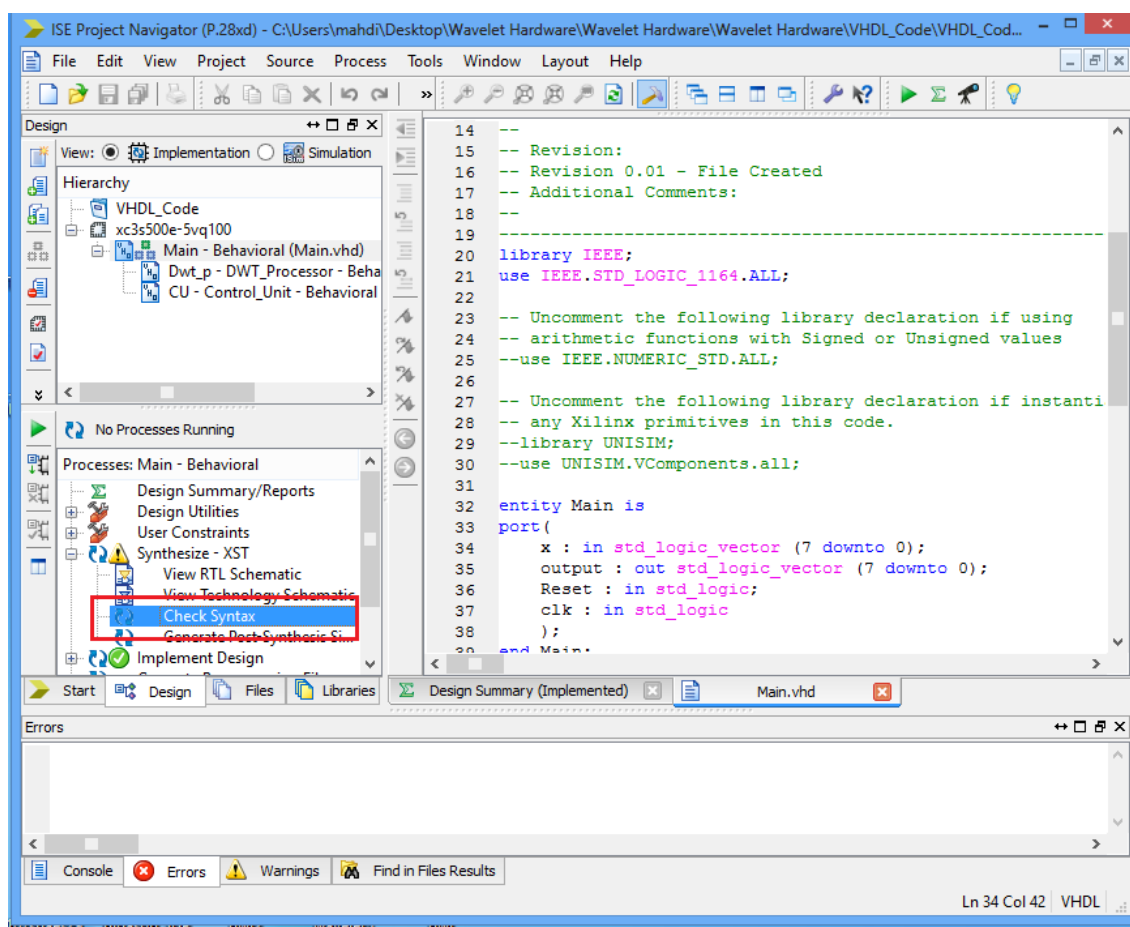
ایجاد یک فایل متنی جدید

برای ایجاد یک فایل متنی جهت نوشتن برنامه VHDL در قسمت Project روی New Source کلیک می کنیم و در پنجره ایجاد شده VHDL Module را انتخاب می کنیم و سرانجام فایل متنی ایجاد می شود.

حال در این فایل می توان برنامه VHDL خود را تایپ و سپس کامپایل کنیم.

بررسی نگارش برنامه VHDL و عملیات سنتز

پس از این که برنامه را تایپ کردید باید مطمئن شویم که برنامه خطای نگارشی ندارد برای این منظور در پنجره فرآیندها از قسمت Synthesize-XST بر روی Check Syntax کلیک می‌کنیم. اگر کنار Check Syntax تیک سبز قرار گرفت یعنی برنامه مشکل نگارشی ندارد در غیر اینصورت خطاها را برطرف کرده و دوباره مرحله قبل را انجام می‌دهیم؛ مانند شکل ۳ که در زیر مشاهده می‌کنید.



شکل ۳: بررسی نگارش برنامه VHDL

در صورتی که فرایند سنتز بدون اشکال به پایان برسد می‌توان نمای RTL تولید شده از طراحی را به صورت گرافیکی مشاهده کنیم. برای این کار کافی است در قسمت Synthesize-XST بر روی

View RTL Schematic کلیک کنیم.

۳- شبیه‌سازی رفتاری

برای شبیه‌سازی هر طرحی، نیاز به یک برنامه تست می‌باشد. این برنامه مانند یک مدار خارجی می‌باشد که سیگنال‌های ورودی لازم برای تست طرح را فراهم می‌آورد.

برای ایجاد برنامه تست، از منوی Project گزینه New Source را انتخاب کنید، پنجره New Source Wizard، باز می‌شود VHDL Test Bench را انتخاب کنید و نام مناسبی را وارد کنید Next و سرانجام Finish را کلیک کنید. سپس برنامه تست ایجاد شده را باز کرده و سیگنال‌های ورودی را مقداردهی اولیه می‌کنیم تا مقادیر سیگنال‌های خروجی مشخص شود. پس از مقداردهی اولیه از بخش Isim Simulator بر روی Behavioral Check Syntax کلیک می‌کنیم تا از عدم وجود خطا در کد تست اطمینان حاصل شود. سپس Simulate Behavioral Model را انتخاب می‌کنیم تا فرآیند شبیه‌سازی آغاز شود.

منابع

- [1] G.Ciprian,“FPGA-based discrete wavelet transform design using matlab/Simulink.”*International Symposium for design and technology of electronic packages* ,2008.
- [2] M. Jeyaprakash, “FPGA Implementation of Discrete Wavelet Transform (DWT) for JPEG 2000,” vol. 2, no. 6, pp. 161–163, 2010.
- [3] M.Leon, “Implementation of the 2-D Wavelet Transform into FPGA for Image,” *Jouanal of physics*,vol. 3,2012
- [4] C. F. Hsieh and T. H. Tsai, “FPGA Implementation of a High-Speed Two Dimensional Discrete Wavelet Transform,” *Appl. Mech. Mater.*, vol. 4, pp. 508–512, Dec. 2013.
- [5] W. Fan and Y. Gao, “FPGA Design of Fast Lifting Wavelet Transform,” *Congr. Image Signal Process.*, no. 2, pp. 362–365, 2009.
- [6] P. Yu, S. Yao, and J. Xu, “An Efficient Architecture for 2-D Lifting-based Discrete Wavelet Transform,” vol. 4, no. 1, pp. 3667–3670, 2010.
- [7] M. O. F. Efficient, R. For, and F. Point, “FPGA Implementation OF Efficient VLSI a Rchitecture for fixed point 1-D DWT using,” vol. 3, no. 4, pp. 37–49, 2013.
- [8] T. Dhanya, “Design and Implementation of Pipeline Architecture for High Performance 2-D Daubechies Wavelet Transform,” vol.4, no. 24, pp. 11–15, 2012.
- [9] N. M. Basheer and M. M. Mohammed, “Design and FPGA Implementation of a Lifting Scheme 2D DWT Architecture,” *Int. J. Recent Technol. Eng.*, vol. 2, no. 1, pp. 34–38, 2013.
- [10] T. A. Saidani, M. O. Atri, Y. A. S. Aid, and R. A. Tourki, “Fpga Real Time Acceleration for Discrete Wavelet Transform of the 5 / 3 Filter for Jpeg 2000 Standard,” *Int. J. Embed. Syst. Appl.*, vol. 2, no. 1, pp. 1–14, 2013.
- [11] H.Malaver, “The LOT:transform coding without blocking effects,” *IEEE Trans. Acoust.Speech Signal Process*, vol. 4, no. 37, pp. 553–559, 1989.
- [12] J.Shapiro, “Embedded image coding using zero trees of wavelet cofficients,” *IEEE Trans. Signal Process*, vol. 4, no. 12, pp. 3445–3462, 1993.
- [13] D.Legall, “Subband coding of images using symmetric short Kernel filters and arithmetic coding techniques,” *IEEE Int. Conf. Acoust. Speech Signal Process.*, pp. 761–764, 1998.

- [14] I. Daubechies, “The wavelet transform. Time frequency Localization and signal analysis,” *IEEE Trans. Theory*, vol. 5, no. 36, pp. 961–1005, 1990.
- [15] S.mallat,“A theory of multiresolution signal decomposition: the wavelet representation,” *IEEE Trans. Pattern Anal*, vol. 7, no. 11, pp. 674–693, 2003.
- [16] M.ghanbari, “Standard codecs image compression to advanced video coding.” 2011.
- [17] W. Sweldens, “The lifting scheme: A custom-design construction of biorthogonal wavelets,” 2002.
- [18] W. Sweldens, “The lifting scheme: A construction of second generation wavelets,” *Ind. Math. Initiat. Univ. South Carolina*, 2001.

Abstract

Wavelet transform is a useful tool for various applications in signal processing And in many fields such as audio and video processing, image noise reduction, pattern recognition, image coding and Compression is used. For this reason providing strategies for increasing the speed of implementation of wavelet transform is considered Which is one of the best strategies is the use of parallel processing and FPGA chips.

A new and effective architecture has been proposed in this thesis for implementation of two dimensional discrete wavelet transform to compress images. In this research lifting schema approach has been used which is a newer and more efficient against filter banks approach, also the 5.3 wavelet filter has been used which is applied for lossless image compressing. The goal of this research is hardware tools complexity reduction, optimization of number of applied components and operation frequency increment.

The new architecture consists of a processor unit for calculation of two dimensional discrete wavelet transform coefficients and one control unit for controlling of data flow in processor and producing the address buses of memory and one external memory for coefficients saving.

This architecture has been simulated with VHDL hardware programming language and Xilinx Ise software which is one of the best and strongest software for FPGA programming. The obtained results has been evaluated and compared with other architecture. Two dimensional discrete wavelet transform with lifting schema approach has been simulated in MATLAB language and output images are compared in two environments.

Keywords: discrete wavelet transform, lifting scheme, (5/3) wavelet filter, Xilinx ISE, filter banks, FPGA, Subband coding



Shahrood University of Technology

Faculty of Electrical Engineering

MSc Thesis in digital Electronics

Implementation of the 2-D Discrete Wavelet Transform on FPGA

BY: Mahdi Khodadadi

Supervisor

Dr.Ali Solimani

Advisor

Dr.Hadi Grailu

September 2016