Architecture Design of 2-D Discrete Wavelet Transformation Algorithm using Field Programmable Gate Array (FPGA)

Maha A. Hasso **Sahla A. Ali**

College of Computer Science and Mathematics
University of Mosul, Mosul, Iraq

Received on: 06/03/2013 **Accepted on: 24/06/2013**

ABSTRACT

In this paper an architecture has been proposed for the 2-D Discrete Wavelet Transform (DWT) and the Inverse Discrete Wavelet Transform (IDWT) based on the Convolution method of the Daubechies 5/3-tap Biorthogonal filter bank in the Algorithm transform to image processing, and implementation it on the FPGA (Field Programmable Gate Array) using VHDL, for benefiting from implementation advantages of these Hardware and save run-time. The processing results proved speed and Efficiency of the proposed architecture, where the employed number of slices is less. So it result to Frequency higher and less run-time.The type of the FPGA based in this paper is Xilinx XC3S500E Spartan-3E using Xilinx ISE 9.2i.

Keywords: 2-D Discrete Wavelet Transformation, Field Programmable Gate Array (FPGA).

تصميم معمارية لخوارزمية تحويل المويجة المتقطع ذي البعدين باعتماد مصفوفة البوابات الحقلية القابلة للبرمجة (FPGA)

مهى عبد الرحمن حسو ***سهلة عبد علي***

قسم علوم الحاسوب

كلية علوم الحاسوب والرياضيات

جامعة الموصل

تاريخ استلام البحث : 2013\3\6 **تاريخ قبول البحث : 2013\6\24**

الملخص

تم في هذا العمل اقتراح معمارية لخوارزمية تحويل المويجة المتقطع ذي البعدين (DWT) ومعكوس تحويل المويجة المتقطع (IDWT) بإعتماد طريقة الالتفاف الرياضي (Convolution) لمرشح دبجي (Daubechies) 5/3 ثنائي التعامد في خوارزمية التحويل لمعالجة الصورة وتطبيقه على مصفوفة البوابات الحقلية القابلة للبرمجة Field Programmable Gate Array (FPGA) باستخدام لغة وصف الكيان المادي (VHDL), للاستفادة من محاسن تطبيق هذه الاجهزة المادية وتوفير وقت التنفيذ. اثبتت النتائج العملية للتطبيق سرعة وكفاءة المعمارية المقترحة من حيث عدد الشرائح المستغلة والتي تكون اقل مما يؤدي الى تردد اعلى وزمن تنفيذ اقل. نوع FPGA المعتمد في هذا العمل هو Xilinx XC3S500E Spartan-3E وباستخدام برنامج Xilinx ISE 9.2i.

الكلمات المفتاحية: تحويل المويجة المتقطع ذي البعدين، مصفوفة البوابات الحقلية القابلة للبرمجة.

1. المقدمة

تم اعتماد تحويلات المويجة لكبس الصورة والذي اصبح سائداً وبشكل واضح اكثر من تحويل جيب التمام المتقطع التقليدي (DCT) وذلك بسبب ادائها الافضل ولأنه يُنجزُ أعلى نِسَب كبس بسبب قدرته في التحليل, كما انه معروف على نحو واسع بامتلاكه ميزات ممتازة لفك الترابط (decorrelation), كما يعرض اداء ممتازاً للكبس بفقدان ودون فقدان. ان الاداء العالي لخوارزمية 2D-DWT في كبس الصورة يُبرر استعماله نواةً في المعيار JPEG2000 لكبس الصور الثابتة [1] والمعيار MPEG-4 texture coding لكبس الصور المتحركة [2]. كما ان التحويل المويجي معروف على نحو واسع بامتلاكه ميزات ممتازة لفك الترابط, وهو يعرض اداءً ممتازاً للكبس بفقدان ودون فقدان, فللحصول على افضل اداء لكبس الصورة يتم الاعتماد على هذا التحويل لتحقيقه شرط اعادة بناء مثالية لذا هذا التحويل مستخدم في معيار JPEG 2000 [3]. أخذ تحويلات المويجة حيزا واسعا باهتمام الكثير من الباحثين وذلك لملاءمتها لكثير من الظواهر ولمختلف المجالات منها معالجة الاشارة والصورة, كبس الصورة, تحليل البيانات, بصمات الاصابع الرقمية, تحسين الصورة, تقليل الضوضاء عن الاشارات وازالتها, وتحليل إشارات الزلازل,... الخ[4]. وبسبب تزايد عدد التطبيقات في مناطق مختلفة, اصبح من الضروري استخدام التطبيق المادي لتحويلات المويجة المتقطع لكفاءته. وفي اي تصميم كفوء يجب ان يؤخذ في الحسبان عدد من السمات مثل الاستهلاك الكهربائي, الطاقة الانتاجية (Throughput),...الخ. وان تقنيات مثل العمليات المتوازية (Parallel Operations) ومبدأ خط الانتاج (Pipelining) تساعد في تحقيق هذه الطلبات [5]. يطبق التحويل المويجي اما باعتماد هيكلية الالتفاف الرياضي (Convolution) او ما يسمى باستجابة النبضة المنتهية (Finite Impulse Response FIR), وتتطلب هذه التطبيقات عدداً كبير من العمليات الحسابية. او اعتماد هيكلية مخطط الرفع (Lifting Scheme) للمويجة.

تم اقتراح العديد من المعماريات لتحويل المويجة في السنوات السابقة[7], ففي المعمارية المقترحة في[8] دمجت التحويل بفقدان ودون فقدان , وفي [9] المعمارية المقترحة خفضت بشكل ملحوظ عدد الضوارب (multiplier) , الجوامع (adder) والسجلات اضافةً الى كمية الوصول للذاكرة الخارجية وأدت الى تقليل كلفة الجهاز واستهلاك القدرة بشكل كفوء في التصميم.

2. خوارزمية تحويل المويجة المتقطع ذي البعدين

يختلف تطبيق خوارزمية تحويل المويجة المتقطع ذات بعدين في VHDL في بعض السمات عن تطبيقه في بيئة MATLAB والاختلاف الرئيس عن الماتلاب هو ان VHDL لا يدعم الكثير من الدوال مثل Convolution، Max، Mod، Flip . لذا ولغرض تطبيق الخوارزمية في VHDL نحتاج الى المعادلات الخطية لتحويلات المويجة المتقطع ذات البعدين. تم في المعماريات المقترحة اعتماد المعادلات الحسابية لخوارزمية تحويل المويجة المتقطع ذات بعدين هي كما يأتي [6]:-

$$X_{LL}^{J}(n_1,n_2) = \sum_{i_1=0}^{k-1}\sum_{i_2=0}^{k-1} g(i_1).g(i_2).X_{LL}^{J-1}(2n_1-i_1)(2n_2-i_2) \quad \ldots\ldots\ldots\ldots\ldots\ldots(1)$$

$$X_{LH}^{J}(n_1,n_2) = \sum_{i_1=0}^{k-1}\sum_{i_2=0}^{k-1} g(i_1).h(i_2).X_{LL}^{J-1}(2n_1-i_1)(2n_2-i_2) \quad \ldots\ldots\ldots\ldots\ldots\ldots(2)$$

$$X_{HL}^{J}(n_1,n_2) = \sum_{i_1=0}^{k-1}\sum_{i_2=0}^{k-1} h(i_1).g(i_2).X_{LL}^{J-1}(2n_1-i_1)(2n_2-i_2) \quad \ldots\ldots\ldots\ldots\ldots\ldots(3)$$

$$X_{HH}^{J}(n_1,n_2) = \sum_{i_1=0}^{k-1}\sum_{i_2=0}^{k-1} h(i_1).h(i_2).X_{LL}^{J-1}(2n_1-i_1)(2n_2-i_2) \quad \ldots\ldots\ldots\ldots\ldots\ldots(4)$$

حيث ان:

J تمثل مستوى تحويل المويجة المتقطع ذات البعدين.

K تمثل طول المرشح.

g(n) استجابة النبضة لمرشح التمرير الواطئ G(x) .

h(n) استجابة النبضة لمرشح التمرير العالي H(x) .

$X_{LL}^{0}(n_1,n_2)$ تمثل الصورة المدخلة.

ان كل مستوى تحليل في DWT يتكون من مرحلتين, وكما مبين في الشكل(1): المرحلة الاولى تمرير المرشح افقياً, والمرحلة الثانية تمرير المرشح عمودياً. في اول مستوى تحليل, حجم الصورة المدخلة هو N×N والناتج من تحليل الصورة للمستوى الاول هو اربع حزم فرعية LL1, LH1, HL1 وHH1 ذات احجام N/2×N/2. في المستوى الثاني من التحليل, الصورة المدخلة هي الحزمة LL1 والناتج من هذا التحليل هو اربع حزم فرعية LL2, LH2, HL2 وHH2 ذات احجام N/4×N/4. في المستوى الثالث من التحليل, الصورة المدخلة هي الحزمة LL2 والناتج من هذا التحليل هو اربع حزم فرعية LL3, LH3, HL3 , وHH3 ذات احجام N/8×N/8. بنفس الاسلوب تم الاستمرار في عملية تحليل الصورة, حجم الحزم الناتجة من التحليل يمكن حسابه من المعادلة الآتية [6]:

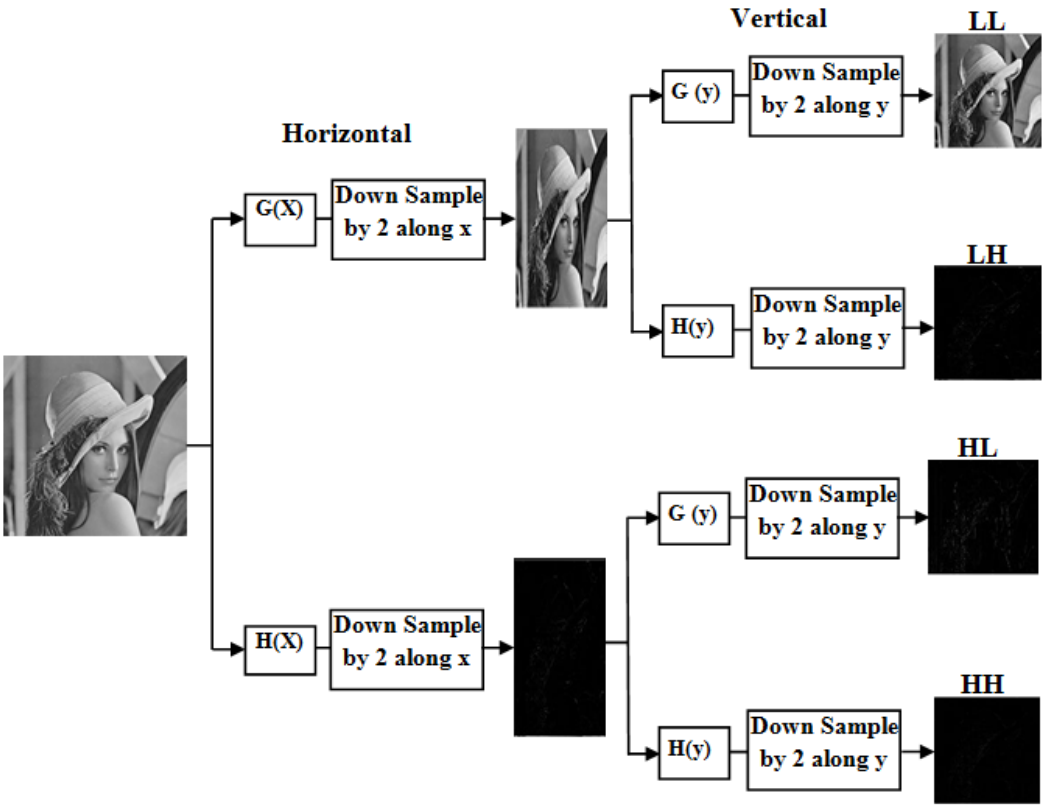
$$\sum_{L=1}^{J} \frac{N}{2^{L-1}} = N + \frac{N}{2} + \frac{N}{2^2} + \frac{N}{2^3} + \cdots + N/2^{J-1} = 2(1-2^{-J})N \quad (5)$$

حيث تمثل

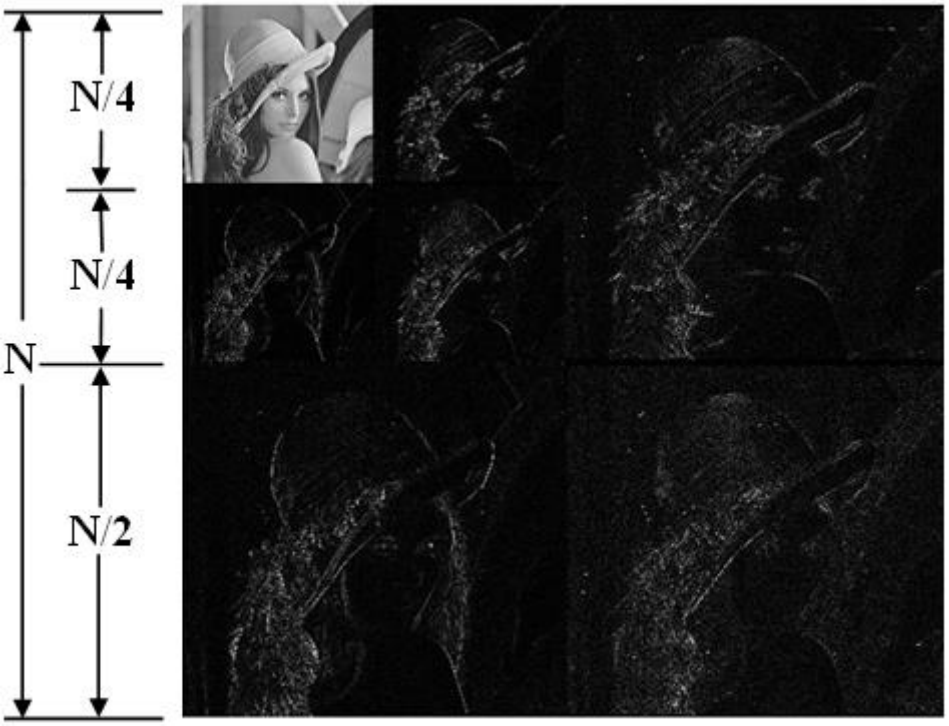
J مستوى التحليل. N حجم البيانات في المستوى الاول.

N/2 حجم البيانات في المستوى الثاني...

$N/2^{J-1}$ الحجم في المستوى J. والشكل(2) يبين نتيجة تحليل صورة "Lena" بعد مستويين من تحليل 2-D DWT .



الشكل(1) تحليل 2-D DWT لمستوى واحد.

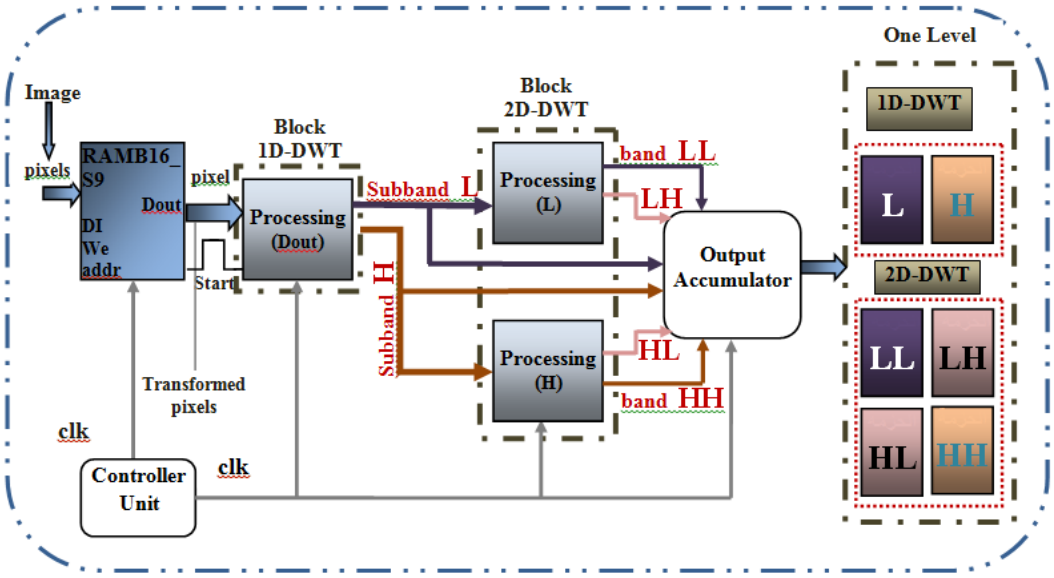


الشكل(2) نتيجة تحليل صورة "Lena" بعد مستويين من التحليل.

3. المعمارية المقترحة لخوارزمية 2D DWT-IDWT باعتماد معمارية الازاحة لتطبيق الالتفاف الرياضي:

i: التطبيق العملي للمعمارية المقترحة لـ 2D DWT الأمامية لمستوى واحد

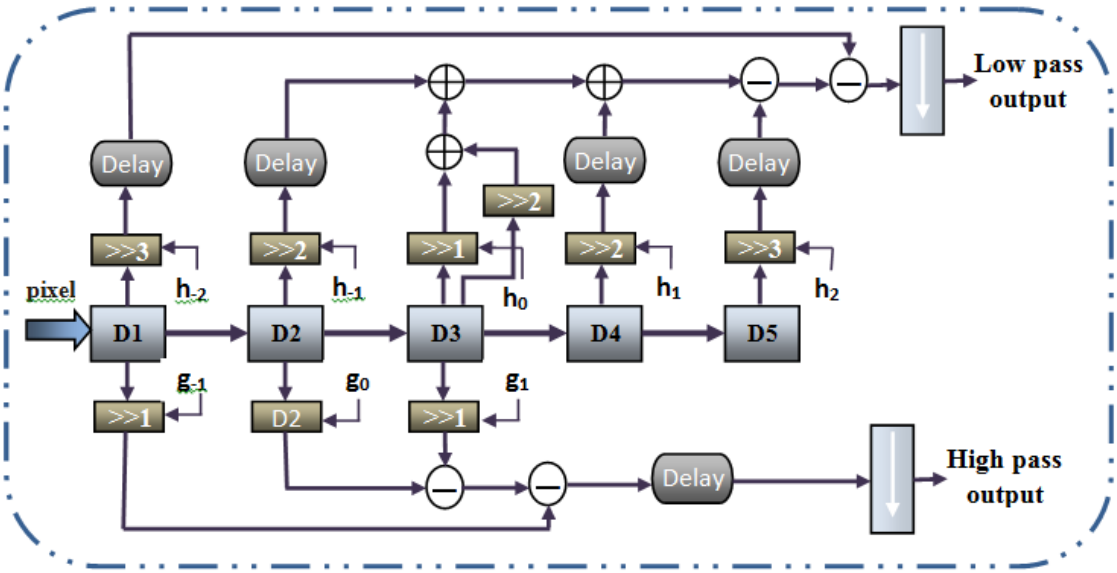
يختلف التعامل مع الصورة في لغة وصف الكيان المادي عنه في لغات اخرى اذ انه لا يمكنه قراءة الصورة بصيغتها القياسية مباشرةً من الحاسوب لذا يجب تغيير صيغة الصورة وتحويلها الى صيغة تفهمها لغة وصف الكيان المادي من هذه الصيغ: السداسي عشر (Hex-Format) او الثنائي (Binary-Format). لذا عند تطبيق خوارزمية التحويل المويجي, تم اولاً تحويل الصورة الى صيغة hex لكي نتمكن من التعامل معها. ان عملية تحويل الصورة من صيغتها القياسية الى صيغة hex تمت ببرنامج في ماتلاب يقرأ الصورة ويخزنها في ملف بصيغة hex. بعد تغيير صيغة الصورة تم خزن قيم الصورة في الذاكرة ومن ثم التعامل مع الصورة من داخل لغة وصف الكيان المادي باستخدام برنامج ISE9.2i والشكل(3) يوضح التطبيق المادي للمعمارية المقترحة لخوارزمية التحويل المويجي المتقطع لمستوى واحد. كما مبين من الشكل ان الذاكرة التي أُستخدمت في المعمارية المقترحة لخزن الصورة هي من نوع RAMB16_S9, وهذا النوع يقراء ويكتب 8-Bit عند كل نبضة, وكما مبين ايضاً انه احتجنا الى ثلاث معالجات لتطبيق 2D-DWT الأمامية لمستوى واحد. ففي المعالج الاول ((Process(Dout) حصلنا على L وH والتي تمثل قيم المعاملات الوسطية الناتجة من إمرار مرشح التمرير الواطئ والعالي افقياً على الصورة اما عند المعالجين ((Process (L و Process (H)) حصلنا على HL,LH,LL وHH الناتجة من إمرار المرشحين عمودياً على L وH.



الشكل(3) الوصف المادي للمعمارية المقترحة لـ2D DWT الأمامية لمستوى واحد.

بما ان طول مرشحي التمرير الواطئ والعالي هو 5 و3 على التوالي, لذا إحتجنا الى خمس مسجلات لخزن قيم البكسلات المقروءة عند كل نبضة لتطبيق عملية الالتفاف الرياضي لكل معاملات المرشحين بوقت واحد, وبهذا العمل نكون قد حققنا Pipelining داخل المعالجات. في المعالج الاول تم عند النبضة الاولى خزن قيمة البكسل الاولى المتكونة من 8-bit في المسجل الاول ومن ثم حساب ناتج الالتفاف الرياضي مابين قيمة البكسل هذه ومعامل التمرير الواطئ (h_{-2}) ومعامل التمرير العالي (g_{-1}) بنفس الوقت تم امرار المرشحين على الصورة, (وبمعنى اخر طُبق الالتفاف الرياضي مع قيمة البكسل مرة باستخدام معامل التمرير الواطئ واخرى باستخدام معامل التمرير العالي في وقت واحد وفي نفس النبضة), وخزن النتيجة في L وH على التوالي. عند النبضة الثانية يتم خزن قيمة البكسل الثانية في المسجل الاول وتمرير قيمة المسجل الاول الى المسجل الثاني ومن ثم

حساب ناتج الالتفاف الرياضي ما بين قيم المسجلين ومعاملي مرشح التمرير الواطئ (h_{-2},h_{-1}) ومعاملي مرشح التمرير العالي (g_{-1},g_0). وعند النبضة الثالثة تم خزن قيمة البكسل الثالثة في المسجل الاول وتمرير قيمة المسجل الاول الى المسجل الثاني و قيمة المسجل الثاني الى المسجل الثالث ومن ثم حساب ناتج الالتفاف الرياضي ما بين قيم المسجلات ومعاملات مرشح التمرير الواطئ (h_{-2},h_{-1},h_0) ومرشح التمرير العالي (g_{-1},g_0, g_{-1}) في وقت واحد وخزن النتيجة في L و H على التوالي, وهكذا تم تطبيق العمل نفسه على كل صفوف الصورة. تم ايضاً في نفس المعالج, بعد حساب نتيجة تمرير معاملات المرشحين افقياً على الصورة, تطبيق عملية الاختزال Down Sample عند كل نبضة على اعمدة الصورة (حذف عمود مابين كل عمودين) حيث يتم خزن قيمة ناتج الالتفاف عند نبضة وترك الثانية عند النبضة التالية وهكذا... ومن ثم يتحقق الاختزال ونحصل على L و H بحجم N×N/2 . تم اعتماد طريقة الازاحة لتطبيق الالتفاف الرياضي في المعمارية المقترحة و الشكل(4) يبين الوصف المادي لعمل هذا المعالج.



الشكل(4) الوصف المادي لعمل المعالج الاول ((Process(Dout)) .

حيث تمثل:

D1 الى D5 المسجلات لخزن قيم البكسلات.

1>> الازاحة الى اليمين بمقدار واحد, حيث ان هذه الازاحة تمثل عملية القسمة لقيمة البكسل على قيمة المعامل 2.

2>> الازاحة الى اليمين بمقدار اثنان, وهي تمثل عملية القسمة لقيمة البكسل على قيمة المعامل 4.

3>> الازاحة الى اليمين بمقدار ثلاث, وهي تمثل عملية القسمة لقيمة البكسل على قيمة المعامل 8.

Delay تمثل التأخير. والشكل يمثل عملية الاختزال على اعمدة الصورة.

($h_{-2},h_{-1},h_0,h_1,h_2$) معاملات مرشح التمرير الواطئ.

($g_{-1},g_0,g_1,$) معاملات مرشح التمرير العالي.

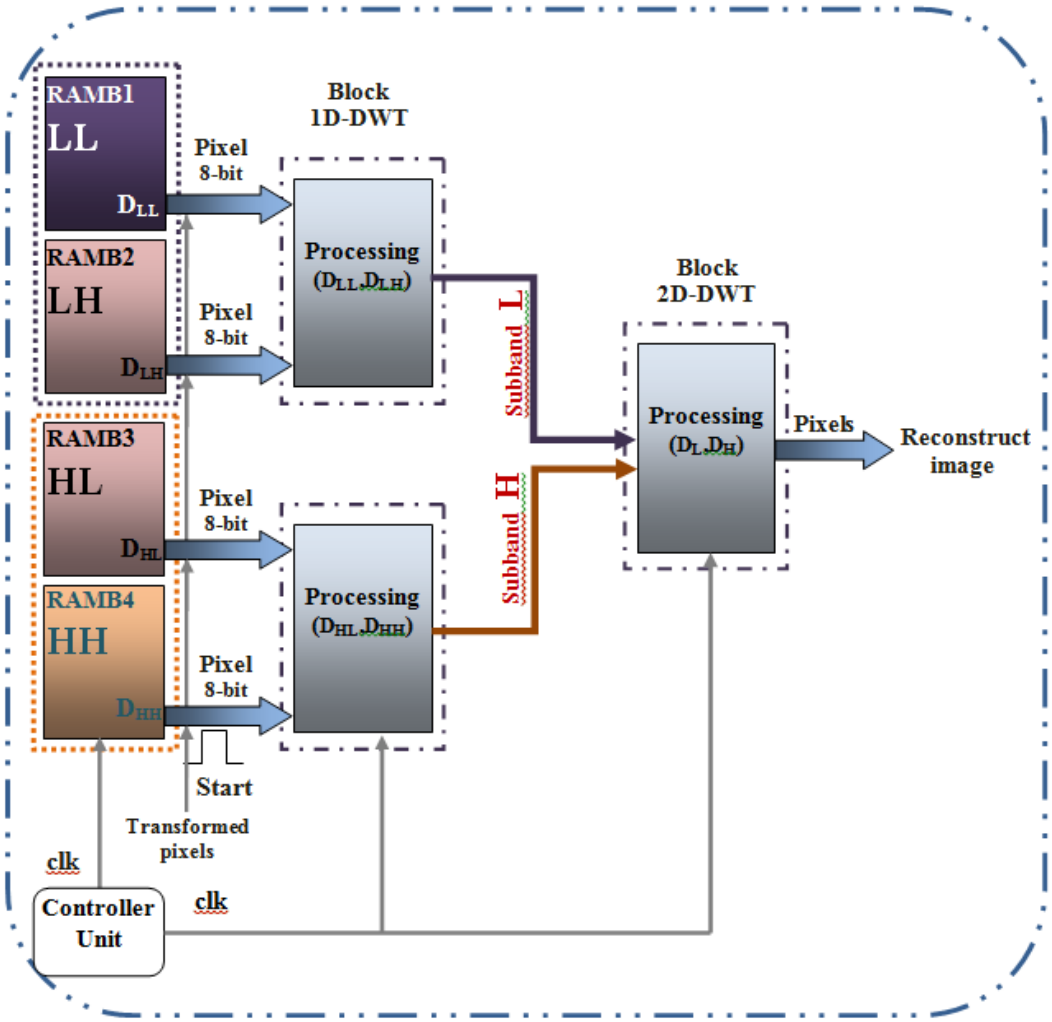
كما مبين من الشكل, تم اضافة تأخير بمقدار نبضة واحدة على التصميم بسبب معامل مرشح التمرير الواطئ h_0 ($(0.11)_2$) 3/4) لانه يحتاج الى نبضتين لحساب ناتج تمرير هذا المعامل على قيمة البكسل للصورة لذا

تم تأخير حساب قيم الالتفاف لباقي معاملات المرشحات بمقدار نبضة واحدة للحصول علي كل نتائج تمرير المعاملات على قيم البكسلات بنفس الوقت وبنفس النبضة وبذلك تكون النتائج صحيحة ودقيقة.

ان اعتماد طريقة الازاحة الى اليمين لتطبيق الالتفاف الرياضي في المعمارية المقترحة والتي تمثل عملية القسمة لقيمة البكسل على قيمة المعامل تعطي ناتج العملية كرقم حقيقي. ان لغة وصف الكيان المادي لا تحتفظ بباقي ناتج القسمة وانما تأخذ فقط الرقم الصحيح من الناتج وتهمل الباقي, لذا لتجاوز هذه الحالة والاحتفاظ بباقي ناتج القسمة قدر الامكان تم بعد كل عملية ازاحة مباشرةً جمع ناتج الازاحة مع قيمة البت المزاحة للبكسل, وللتوضيح اكثر: اذا كانت عملية الازاحة الى اليمين لقيمة البكسل بمقدار واحد ستكون قيمة البت للبكسل التي تجمع مع ناتج الازاحة هي البت صفر (bit0), بينما اذا كانت عملية الازاحة بمقدار اثنين ستكون قيمة البت للبكسل التي تجمع مع ناتج الازاحة هي bit1, واذا كانت عملية الازاحة بمقدار ثلاثة ستكون قيمة البت التي تجمع مع ناتج الازاحة هي bit2 وهكذا..., ان هذه العملية تقوم بإضافة واحد الى ناتج الازاحة اذا كان باقي عملية القسمة اكبر او يساوي خمسة, وبهذا تم الحفاظ على جزء من ناتج القسمة وتقليل نسبة الخطأ للحصول على نتائج اكثر دقة. بعد انتهاء المعالج الاول والحصول على القيم الوسطية L وH يبدأ المعالجان الثاني والثالث بالعمل (Process (L) وProcess (H)) في هذه المرحلة تتحقق خاصية التوازي لان المعالجين يبدأن العمل في نفس الوقت وفي نفس النبضة ويكون عدد البت التي يتم معالجتها هي 16-bit عند اول نبضة. ان طريقة المعالجة التي يتبعها المعالجان الثاني والثالث هي نفس طريقة عمل المعالج الاول ماعدا ان طريقة المرور لمعاملات المرشحين يكون عمودياً على L وH, وعملية الاختزال تكون على صفوف المصفوفة. عند اكمال معالجة قيم المعاملات الوسطية فان الناتج من هذا العمل المتوازي للمعالجين هو اربع حزم فرعية: HL,LH,LL وHH ذات احجام N/2×N/2. بعد اكمال هذه المرحلة نكون قد حققنا التطبيق المادي للمعمارية المقترحة لـ2D DWT الأمامية لمستوى واحد.

ii: التطبيق العملي للمعمارية المقترحة لـ 2D IDWT العكسية لمستوى واحد

يوضح الشكل(5) التطبيق المادي للمعمارية المقترحة لخوارزمية التحويل المويجي المتقطع العكسية لمستوى واحد. كما مبين من الشكل انه احتجنا الى ثلاث معالجات للمعمارية المقترحة العكسية لمستوى واحد, وكما ذكرنا سابقاً ان كل مستوى تحليل في DWT يتكون من مرحلتين, لذا لتطبيق IDWT سيكون هناك مرحلتان ايضاً لاعادة تركيب الصورة. وكما هو مبين في الشكل ان في المرحلة الاولى لاعادة التركيب يوجد معالجان يعملان بنفس الوقت (Process(D_{LL},D_{LH}) وProcess(D_{HL},D_{HH})) اي يوجد حالة التوازي حيث تم التعامل مع اربع ذاكرات في وقت واحد عند كل نبضة, وبذلك يكون عدد البكسلات التي تعالج عند النبضة الاولى هي 4 بكسل (32 bit), بكسل واحدة من كل ذاكرة. في هذه المرحلة تم إمرار مرشح التمرير الواطئ والعالي افقياً على الحزم الاربعة التي حصلنا عليها من DWT الأمامية وسيكون ناتج هذه المرحلة هي L وH والتي تمثل قيم المعاملات الوسطية, اما في المرحلة الثاني فيوجد معالج واحد (Process(D_L,D_H)) وعدد البكسلات التي تم معالجتها عند النبضة الاولى هي 2 بكسل (16 bit) وتم في هذه المرحلة إمرار المرشحين عمودياً على L وH الناتج من المرحلة الاولى, ناتج هذه المرحلة هي الصورة التي تم تركيبها من ناتج المرحلتين.

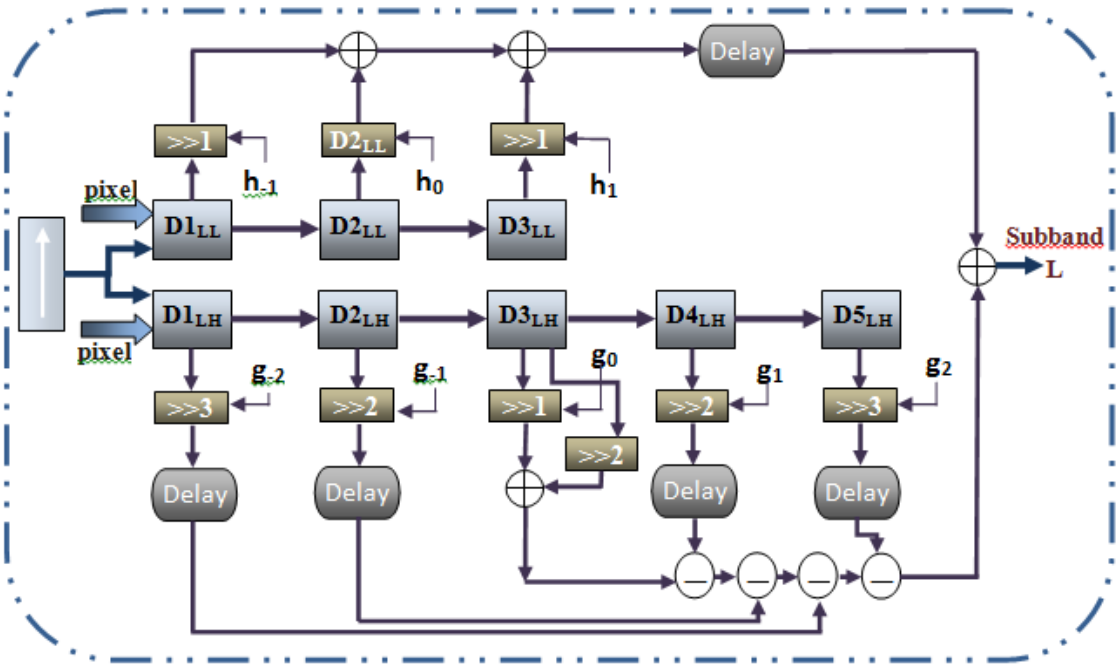


الشكل(5) الوصف المادي للمعمارية المقترحة لـ2D-IDWT العكسية لمستوى واحد.

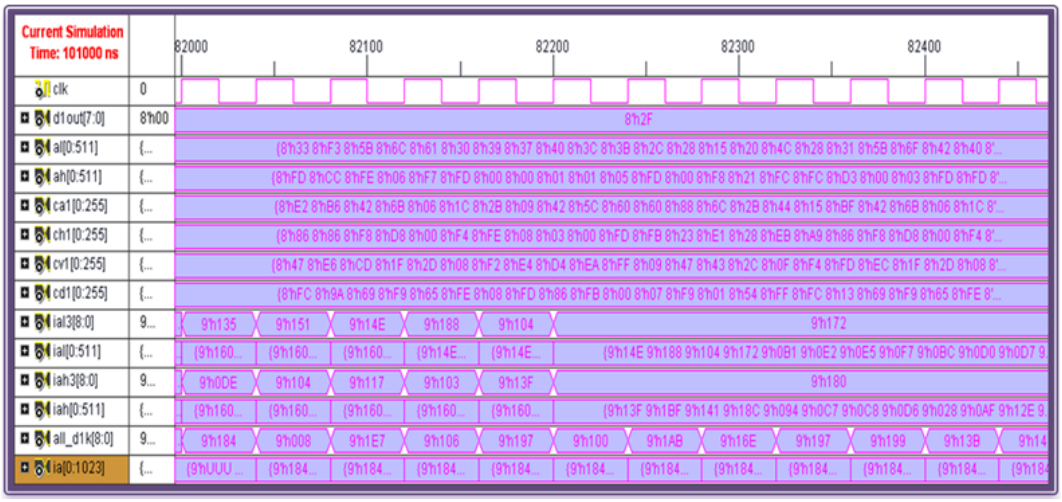
ان طريقة المعالجة المتبعة في هذه المعمارية هي كما يأتي: في المعالج الاول (Process(D_{LL},D_{LH})) نحتاج الى ثمان مسجلات لخزن قيم البكسلات المقروءة عند كل نبضة من الذاكرتين RAMB1 و RAMB2 لتطبيق عملية الالتفاف الرياضي لكل المعاملات للمرشحين بوقت واحد, وبهذه الطريقة يتحقق Pipelining داخل المعالج.

ان الشكل (6) يبين بوضوح طريقة عمل هذا المعالج, كما يبين ايضاً انه تم استخدام ثلاث مسجلات لخزن قيم البكسلات المقروءة من الذاكرة RAMB1 وخمسة مسجلات لخزن قيم البكسلات المقروءة من الذاكرة RAMB2 عند كل نبضة. فعند النبضة الاولى تم قراءة البكسل الاولى ذات حجم 8-bit من كل من RAMB1وRAMB2 وخزنها في مسجلين, وتم بعدها حساب ناتج الالتفاف الرياضي مابين قيمة البكسل لذاكرة RAMB1 ومعامل التمرير الواطئ (h_{-1}) وبين قيمة البكسل لذاكرة RAMB2 ومعامل التمرير العالي (g_{-2}) بنفس الوقت وخزن النتيجة في L وH على التوالي. عند النبضة الثانية تم تطبيق عملية Up Sample وهي عملية اضافة عمود صفري, حيث تم خزن القيمة صفر في المسجل الاول التابع لكل من RAMB1 وRAMB2 وتمرير قيم المسجلين الى المسجل الثاني التابع لكل من الذاكرتين, ومن ثم حساب ناتج الالتفاف الرياضي ما بين قيم المسجلين لـ RAMB1 ومعاملي مرشح التمرير الواطئ (h_0,h_{-1}) وبين قيم المسجلين لـ RAMB2 ومعاملي

مرشح التمرير العالي (g_{-1},g_0) وخزن النتيجة في L و H على التوالي. وعند النبضة الثالثة تم خزن قيمة البكسل الثانية من كل من الذاكرتين وخزنهما في المسجل الاول التابع لكل منها وتمرير قيمة المسجل الاول الى المسجل الثاني وقيمة المسجل الثاني الى المسجل الثالث التابع للذاكرتين ومن ثم حساب ناتج الالتفاف الرياضي ما بين قيم المسجلات لذاكرة RAMB1 ومعاملات مرشح التمرير الواطئ (h_{-1},h_0, h_{-1}) وبين قيم المسجلات لذاكرة RAMB2 ومرشح التمرير العالي (g_{-2},g_{-1},g_0) في وقت واحد وخزن النتيجة في L و H على التوالي. عند النبضة الرابعة تم تطبيق عملية Up Sample سيكون طريقة حساب الالتفاف بنفس الطريقة السابقة, وهكذا يُكرر نفس العمل السابق على كل صفوف الصورة. مما سبق يتبين انه تم تطبيق الالتفاف الرياضي وعملية Up Sample في نفس الوقت وعند نفس المعالج.ان طريقة المعالجة التي يتبعها المعالج الثاني هي نفس طريقة المعالج الاول حيث ان المعالجان يعملان بالتوازي في نفس الوقت وعند نفس النبضة.اما بالنسبة الى المعالج الثالث (Process(D_L,D_H)) في معمارية التحويل الموجي العكسي, فان طريقة المعالجة وحساب الالتفاف الرياضي التي يتبعها هي نفس طريقة عمل المعالجين ماعدا ان عدد البت التي تمت معالجتها هي 16-bit عند النبضة الاولى, وطريقة المرور لمعاملات المرشحين يكون عمودياً على الحزم L و H الناتجة من المعالجين, وطبقت عملية Up Sample على الاعمدة أي تم اضافة عمود صفري. عند اكمال عملية المعالجة لقيم المعاملات الوسطية يتم استرجاع الصورة. لقد تم ايضاً في هذه المعمارية اضافة تأخير بمقدار نبضة واحدة على التصميم بسبب معامل مرشح التمرير العالي g_0 ($(0.11)_2$) (3/4) لانه يحتاج الى نبضتين لحساب ناتج تمرير هذا المعامل على قيمة البكسل لذا تم تأخير حساب قيم الالتفاف لباقي معاملات المرشحات بمقدار نبضة واحدة للحصول على كل نتائج تمرير المعاملات على قيم البكسلات بنفس الوقت وبنفس النبضة لضمان الحصول على نتائج صحيحة. بعد اكمال هذه المرحلة تم تحقيق التطبيق المادي للمعمارية المقترحة لـ2D IDWT العكسية لمستوى واحد, والشكل(7) يبين ناتج التطبيق المادي Simulation للمعمارية المقترحة لـ2D-DWT& IDWT لمستوى واحد.



الشكل(6) الوصف المادي لعمل المعالج الاول (Process(D_{LL},D_{LH})) .

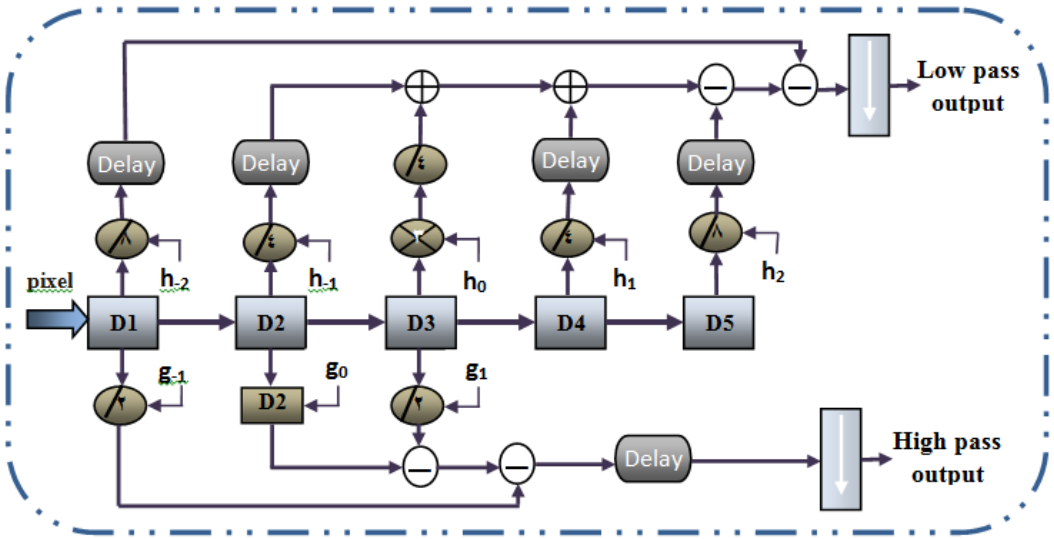


الشكل(7) نتائج Simulation للمعمارية المقترحة لـ2D-DWT& IDWT لمستوى واحد.

4. المعمارية المقترحة لخوارزمية 2D DWT-IDWT باعتماد معمارية القاسم لتطبيق الالتفاف الرياضي

i: التطبيق العملي للمعمارية المقترحة لـ 2D DWT الأمامية لمستوى واحد

لقد تم تصميم المعمارية لخوارزمية التحويل الموجي ذات البعدين ايضاً باعتماد طريقة القاسم لتطبيق الالتفاف الرياضي. ان التطبيق المادي للمعمارية المقترحة باعتماد القاسم لمستوى تحليل واحد مشابه للمعمارية المقترحة باعتماد الازاحة كما هو موضح في الشكل (3) السابق. اما الشكل(8) يبين الوصف المادي لعمل المعالج الاول في التحويل الموجة الامامي, حيث ان طريقة عمل هذا المعالج هي نفس طريقة المعالجة للمعالج الاول في المعمارية المقترحة باعتماد الازاحة ماعدا ابدال عمليات الازاحة بعمليات القسمة كما هو مبين من الشكل.

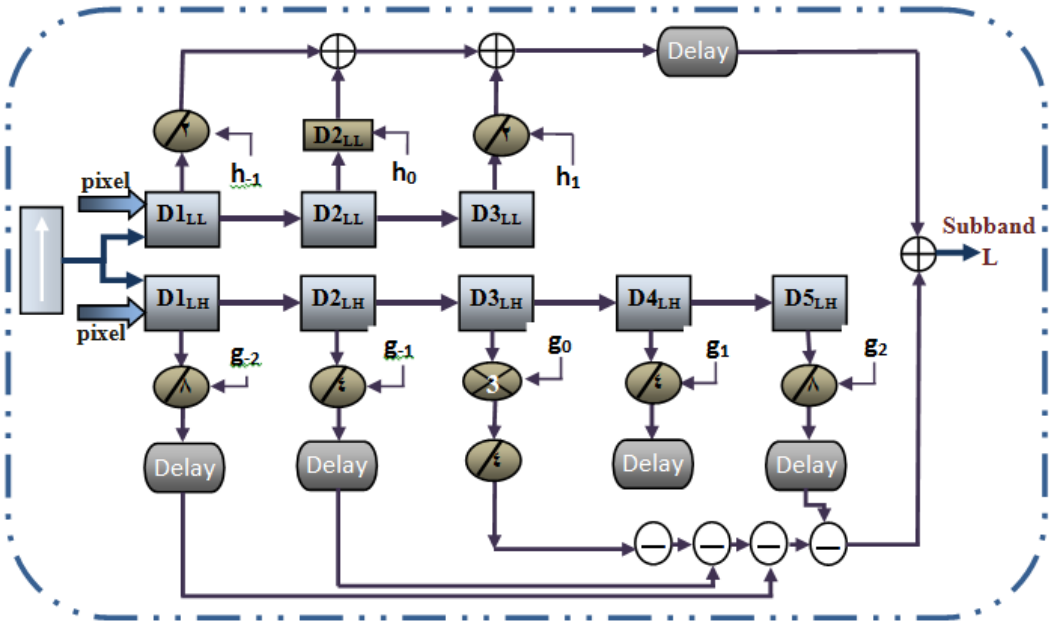


الشكل(8) الوصف المادي لعمل المعالج الاول للمعمارية المقترحة لـDWT باعتماد القاسم.

ii: التطبيق العملي للمعمارية المقترحة لـ 2D IDWT العكسية لمستوى واحد

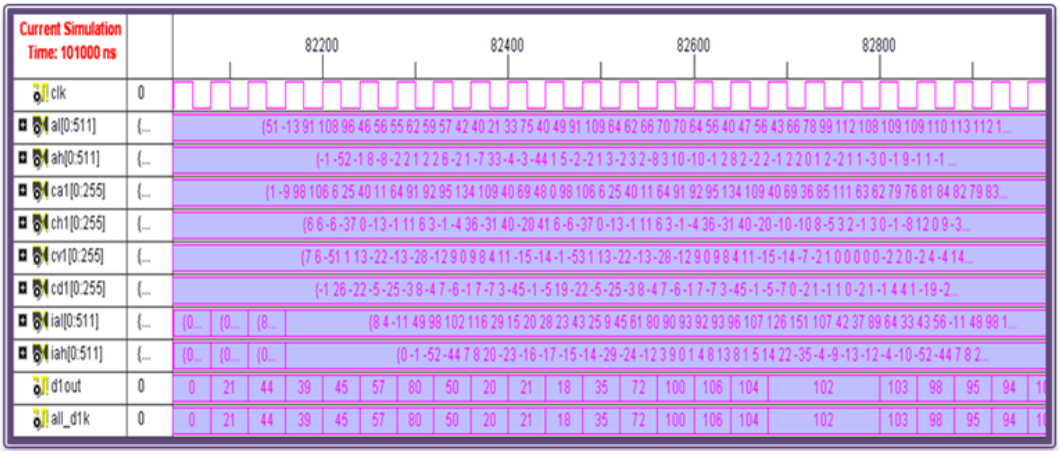
ان المعمارية العكسية المقترحة باعتماد القاسم مشابه للمعمارية المقترحة باعتماد الازاحة الموضحة في الشكل (5) السابق, كما ان طريقة عمل المعالجات في هذه المعمارية هي نفس طريقة العمل في المعمارية

IDWT المقترحة باعتماد الازاحة ماعدا ابدال عمليات الازاحة بعمليات القسمة. الشكل(9) يبين الوصف المادي لعمل المعالج في المعمارية العكسية المقترحة باعتماد القاسم.



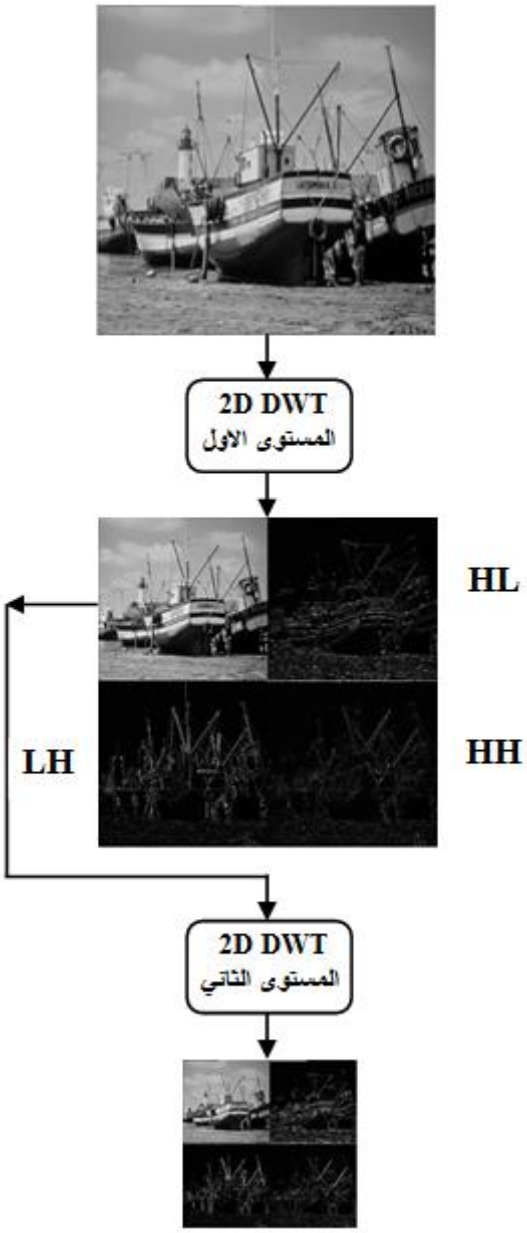
الشكل(9) الوصف المادي لعمل المعالج الاول للمعمارية المقترحة لـIDWT باعتماد القاسم.

ان الشكل(10) يبين ناتج التطبيق المادي Simulation للمعمارية المقترحة لـIDWT& 2D-DWT لمستوى واحد باعتماد طريقة القاسم لتطبيق الالتفاف.



الشكل(10) نتائج Simulation للمعمارية المقترحة لـIDWT& 2D-DWT لمستوى واحد باعتماد القاسم.

ان الشكل (11) يوضح الصورة الأصلية والصور الثانوية لمستويين الناتجة من تنفيذ التحويل الموجي الأمامي وكذلك الصورة المسترجعة. أما الشكل (12) يبين عملية تحليل الصورة Boat لمستويين.



الشكل(11) تحليل الصورة Boat لمرشح 5/3 لمستويين.

الشكل(12) تحليل واسترجاع الصورة Boat لمرشح 5/3 .

5. الاستنتاجات:

ان وجود عملية خط الانابيب ضمن المعماريات المقترحة تؤدي الى زيادة عنصر المعالجة عند كل نبضة, فمثلاً في معمارية تحويل المويجة المتقطع الامامي عند النبضة الاولى يوجد بكسل واحدة تم معالجتها لكن في النبضة الثانية يزداد عدد البكسلات الى اثنين وهكذا يزداد العدد عند كل نبضة الى مابعد النبضة الخامسة حيث يستقر العدد الى خمسة بكسلات التي يتم معالجتها عند كل نبضة (اي ما بعد النبضة الخامسة), وهذا يعني زيادة الطاقة الانتاجية (Throughput). تعد الطاقة الانتاجية عاملاً مهماً للعديد من التطبيقات والذي يقصد به زيادة عدد البكسلات او العينات التي يتم معالجتها في وقت واحد وعند كل نبضة, وهذا يؤدي الى اداء افضل.كما ان وجود خاصية التوازي ضمن المعماريات المقترحة تؤدي الى تضاعف الطاقة الانتاجية ومن ثم زيادة عملية المعالجة عند كل نبضة, لذا ان وجود عملية خط الانابيب مع خاصية التوازي ضمن المعماريات المقترحة يؤدي الى معالجة كفوءة وأداء افضل.

ان تطبيق المعمارية المقترحة لخوارزمية 2D DWT-IDWT عند اعتماد معمارية الازاحة في المعمارية المقترحة فإنه يستهلك عدد من الشرائح او الدوائر المنطقية من المنطقة ضمن شرائح (Slices) الجهاز, لكن عند اعتماد معمارية القاسم فإنه يستهلك فوائد الضوارب المخصصة الموجودة في الجهاز. الجدول (1) يبين المقارنة بين المعماريتين المقترحتين لخوارزمية التحويل المويجي باعتماد معمارية الازاحة والقاسم, تشير المقارنة ان عدد الشرائح المستغلة تكون اقل عند اعتماد معمارية الازاحة والذي يؤدي الى تردد اعلى وزمن تنفيذ اقل.

الجدول (1) المقارنة بين نتائج تطبيق المعماريتين المقترحتين لخوارزمية 2D-DWT& IDWT باعتماد الازاحة والقاسم لمستوى واحد.

المعمارية المقترحة	المصادر	العدد المستغل	العدد الكلي	نسبة الاستخدام
المعمارية المقترحة لـ-2D DWT& IDWT باعتماد الازاحة لمستوى واحد	Slices	1085	4656	%23
	Slice Flip Flops	1190	9312	%12
	4 Input LUTs	1911	9312	%20
	Bonded IOBs	9	232	%3
	Multipliers	0	20	%0
	Min period	11.18		
	Max Frequency	89.445		
المعمارية المقترحة لـ-2D DWT& IDWT باعتماد القاسم لمستوى واحد	Slices	1179	4656	%25
	Slice Flip Flops	1163	9312	%12
	4 Input LUTs	1935	9312	%20
	Bonded IOBs	33	232	%14
	Multipliers	6	20	%30
	Min period	12.472 ns		
	Max Frequency	80.180 MHz		

المصادر

[1] ISO/IEC FCD15444-1, (2000), "JPEG 2000 Image Coding System".

[2] ISO/IEC JTC1/SC29/WG11, FCD 14496-1, (1998), "Coding of Moving Pictures and Audio".

[3] Guoan Yang & Huub Van de Wetering & Songjun Zhang, (2012), "Optimization Design of Biorthogonal Wavelet Filter Banks for Extending JPEG 2000 Standard Part-2 ", J Sign Process Syst.

[4] Gaurav Tewari, Santu Sardar, K. A. Babu,(2011)," High-Speed & Memory Efficient 2-D DWT on Xilinx Spartan3A DSP Using Scalable Polyphase Structure with DA for JPEG2000 Standard", IEEE,.

[5] D. U. Shah1, C. H. Vithlani2 , (2011),"Efficient Implementation of Discrete Wavelet Transforms Using FPGAs ", International Journal of Advances in Engineering & Technology, Sept.

[6] Po-Cheng Wu and Liang-Gee Chen, (2001)," An Efficient Architecture for Two-Dimensional Discrete Wavelet Transform", IEEE Transactions On Circuits And Systems For Video Technology, Vol. 11, No. 4, April.

[7] Dhaha Dia, Medien Zeghid, Taoufik Saidani, Mohamed Atri, Belgacem Bouallegue,Mohsen Machhout and Rached Tourki,(2009),"Multi-level Discrete Wavelet Transform Architecture Design ", Proceedings of the World Congress on Engineering, Vol I, WCE 2009, July 1 - 3, London, U.K. ,.

[8] M. Martina and G. Masera,(2007),"Multiplierless, Folded 9/7-5/3 Wavelet VLSI Architecture", IEEE Transactions on Circuits and Systems- II: Express Briefs, vol.54 No. 9, September.

[9] X. Chengyi, T. Jinwen and L. Jian ,(2006)," Low complexity reconfigurable architecture for the 5/3 and 9/7 discrete wavelet transform", Journal of Systems Engineering and Electronics vol. 17 No.2, pp. 303-308,.