تطوير نظام معالجة تفرعي للتعرف على إيماءات الصم والبكم وتطبيقه بالاعتماد على FPGA

د.م. عمار زقزوق

أستاذ مساعد في كلية الهندسة الميكانيكية والكهربائية – جامعة البعث

ملخص البحث

في هذا البحث، قمنا بمعالجة مصفوفة تعبر عن صورة اليد البشرية بغية الحصول على مميزات هذه الصورة. في سبيل تحقيق ذلك، استخدمنا تقنية FPGA[1]، من خلال تقسيم عمليات المعالجة إلى ثلاثة مسالك، يتم تنفيذها على التفرع. إذ ينفذ كل مسلك باستخدام التقنية الأنبوبية بتقسيمه إلى ثلاثة مقاطع. بعد ذلك، عملنا على تقدير التسريع الذي حصلنا عليه نتيجة استخدامنا للتقنية الأنبوبية والمسالك التفرعية. وبالتالي، أصبح لدينا إمكانية تصميم نظام مضمن داخل شريحة، واستخدام الهواتف النقالة كأجهزة متكاملة من ناحية الموارد المادية والبرمجية.

كلمات مفتاحية: إيماءات الصم والبكم – معالجة الصور – تعرف على الأشكال – معالجة أنبوبية – مسالك – معالجة تفرعية – دارات قابلة للبرمجة – لغة توصيف مادية – نظام مضمن – نظام ضمن شريحة – تسريع.

[1] Field Programmable Gate Array: مصفوفة البوابات القابلة للبرمجة حقلياً.

1- المقدمة (Introduction):

تختلف لغة الإشارة من بلد إلى آخر ومن جهة إلى أخرى، حتى أنها تختلف ضمن المنطقة الواحدة، وذلك يعود إلى الاصطلاحات الخاصة بين الصم أنفسهم، وهذا يؤدي إلى تشعب الأمر بشكل كبير وتعقده.

أول من بادر إلى تنظيمها وتقنينها هو الأب (دولابي) الذي نظم الإشارات التي يستعملها الصم ودونها في قاموس صغير، وأصبحت هذه اللغة هي اللغة الأساسية في المدارس التي كان يشرف عليها. ومن بين من شارك في نشر هذه اللغة غالوديه الذي سافر سنة 1817 إلى أمريكا، وأسس مدرسة لتعليم الصم تحمل إلى اليوم اسمه بعدما تطورت، إلى أن أصبحت اليوم أول جامعة في العالم تعتني بالتعليم العالي للصم والبحوث والدراسات ويرأسها عميد أصم، ويشكل الصم نسبة عالية من الأساتذة، وتعتمد فيها لغة الإشارة بالدرجة الأولى [1].

ما دعا إلى بعض الاتفاقيات بين البلدان على توحيد هذه اللغة بينهم، طبعاً على أساس اللغات المنطوقة. فمثلاً، هنالك لغة الصم والبكم الأميركية (American Sign Language: ASL)، وهنالك لغة الصم والبكم الفرنسية (French Sign Language: FSL). كما تم إعداد أبجدية الإيماءات العربية المقدمة للاتحاد العربي للهيئات العاملة في رعاية الصم ضمن الفترة 1982-1984.

إن الإيماءات في اللغة الواحدة قد تتجاوز 5000 إيماءة، ولا يقتصر ذلك على اليد الواحدة أو كلتا اليدين، بل يتعدى ذلك ليشمل الرأس والجسم بكامله، وهذا يؤدي إلى التعقيد الزائد، ولكن إلى الدقة في إيصال المعلومة بين الصم.

يكفي أن نذكر أن لغة الصم والبكم تتحسن بشكل مستمر لتواكب التطور بمقابلها باللغة المنطوقة [2]. إن التطورات الأخيرة ساعدت في المجالين التربوي والتقني على زيادة الاهتمام بتقديم برامج تتناسب مع قدرات الطلبة المعاقين، وأضحى الحاسوب وسيلة مهمة للتغلب على العقبات التي كانت تحول دون استقلاليتهم. كما أنه يساعد على نقل بعض الظواهر الحقيقية للطلبة الصم الذين يعتمدون على حاسة البصر أكثر، وخاصة الظواهر التي يصعب مشاهدتها لبعدها المكاني أو لندرة حدوثها في بيئتهم [3][4].

أشارت الدراسات التربوية إلى أن أول استخدام للحاسوب في مجال التربية والتعليم

لذوي الإعاقة السمعية كان عام 1970 من قبل المكتب التربوي الأميركي [5].

في أحدث تجربة علمية في كلية العلوم بجامعة المنصورة، قام الأستاذ الدكتور أحمد طلبة من قسم الفيزياء بإجراء تجربة لجهاز جديد يساعد الصم والبكم في التغلب على المشكلات التي يعاني منها هؤلاء في تلقي المعلومة، ويمكنهم من الوسيلة الأنجع لمساعدتهم على الخروج من عزلتهم ودمجهم بشكل كامل في المجتمع عن طريق استخدام قفاز يترجم لغة الإشارة [6].

نفذت دراسة في جامعة الموصل، إذ تم استخدام المقدر اللبي (Kernel Estimator) وأسلوب العنقدة بمعدل k للتعرف على إيماءة اليد، وكتابة البرنامج باستخدام ماتلاب [7].

وفي بحث قريب، تم تصميم نظام ANFIS لتمييز اشارة يد أو يدين للغة الإشارة في الهند في جامعة مومباي [8].

وبخصوص البنى المادية والبرمجية التي استخدمت في مجال المعالجة التفرعية للصور، فقد تم إجراء العديد من الأبحاث، نذكر منها الآتي:

تم نشر بحث في مجال معالجة الصور باستخدام الحوسبة التفرعية ضمن بنية متعددة النوى وتطبيقاته في مجال الصور الطبية [9].

وفي بحث آخر، تم إجراء تحليل كفاءة الحوسبة التفرعية في مهام معالجة الصور باستخدام تقنية تمرير الرسائل [10].

وكذلك، تم نشر بحث طبقت فيه بنية نظام معالجة صور باستخدام FPGA من أجل أنظمة الرؤية المضمنة [11].

سنعمل في نهاية بحثنا على مقارنة النظام الذي قمنا بتطويره مع هذه الأبحاث.

2- مشكلة البحث (Research Problematic):

تمثل فئة الصم والبكم شريحة لا تتجزأ من المجتمع، ولغة الإشارة هي لغة خاصة بينهم، ويواجه الناس الطبيعيون مشكلة كبيرة في التواصل معهم، إذ أن الإنسان الطبيعي لا يحتاج إلى تعلم لغة الإشارة. لذلك، فإن فئة الصم والبكم معزولة نوعاً ما عن الناس الطبيعيين. وهنا، يجب علينا التدخل وإدخال هذه الفئة المهمة إلى داخل المجتمع، لأنهم

أناس طبيعيون ولكن صلة الوصل معهم مقطوعة، وسنعمل على إيجاد هذه الصلة.

3- أهمية البحث (Research Importance):

تكمن أهمية هذا البحث في أن إيجاد مترجم للغة الإشارة إلى اللغة الشفوية سيساعد الناس الصم والبكم على الانخراط في المجتمع والتواصل مع محيطهم العام، وبالتالي الاستفادة منهم في العديد من الأعمال وحتى الدراسات والاختراعات.

4- هدف البحث (Research Aim):

يهدف هذا البحث إلى الآتي:

1- تمييز إيماءات لغة الصم والبكم، وترجمة كل حركة بلغة الإشارة إلى اللغة الشفوية.

2- إيجاد صلة وصل بين الصم والبكم والناس الطبيعيين.

3- توجيه النظر لذوي الاحتياجات الخاصة، ومحاولة مساعدتهم في التغلب على مشكلاتهم.

5- فرضيات البحث (Research Thesis):

يتم الافتراض في هذا البحث أننا نتعامل مع لغة واحدة من لغات الصم والبكم فقط، مع العلم أن هناك العديد من لغات التواصل بين الصم والبكم الموجودة، وهذه اللغة هي لغة الإشارة بيد واحدة أو باليدين. وقد اعتمدنا في بحثنا هذا على لغة الصم والبكم العربية، المتفق عليها من قبل جامعة الدول العربية، وقمنا بالتركيز بشكل أساسي على تمييز الأحرف الأبجدية.

6- منهج البحث (Research Methodology):

يعتمد البحث على المنهج التطبيقي من خلال تطبيق إشارات اليد كمصفوفة ضمن FPGA.

7- عينة البحث (Research Sample):

يبين الشكل (1) عينة البحث التي تشكل الهدف الأساسي له.

الشكل (1): عينة البحث.

8- مجال تطبيق البحث (Research Application Domain):

يمكن استخدام مثل هذا التطبيق من أجل بناء النظم المضمنة التي يمكنها أن تعمل بمفردها، لتحقيق هدف معين، وهو في حالتنا هذه تمييز إيماءات الصم والبكم، وذلك بسرعة وفعالية جيدة.

الأمر الأساس في هذا البحث هو الحصول على تسريع كبير نوعاً ما، من خلال تنفيذ خوارزمية معالجة الصورة بشكل تفرعي. إذ أن المشكلة في هذه الخوارزمية كانت تتمثل في التأخير الحاصل في برنامج معالجة الصورة. وبالتالي، فإن التسريع هو أمر مطلوب من أجل دخول مثل هذه الخوارزميات حيز العمل في النظم المضمنة والأنظمة ضمن شرائح [12][13][14][15].

9- أساسيات البحث (Research Basics):

يتم التعامل مع الصورة من قبل الحاسوب على أنها مصفوفة ثنائية البعد بالنسبة للصورة ذات السوية الرمادية، وثلاثية البعد من أجل الصورة الملونة. إذ يتكون البكسل في حالة الصورة الملونة من 3 مركبات هي: مركبة اللون الأحمر والأخضر والأزرق. والعمل

على معالجة الصورة هو عملية معالجة على المصفوفة التي تمثل هذه الصورة. لذلك، يعتبر تسريع العمليات على المصفوفات وبناء معالجات متخصصة للعمل عليها أمراً بالغ الأهمية في تطوير مجال معالجة الصور [16].

قبل البدء بالدراسة التفصيلية لهذا البحث، سنذكر أولاً خطوات عمل الخوارزمية التسلسلية في استخلاص ميزات الصورة التي تتضمن حركة اليد البشرية وفق الآتي:

1- الحصول على المسقط اليساري لليد في الصورة، وذلك بتنفيذ العمليات الآتية على الأسطر من الأول حتى الأخير:

أ- جلب بكسل جديد من الذاكرة.

ب- فحص البكسل، فإذا كان البكسل الحالي ينتمي إلى اليد، نفذ الخطوتين التاليتين، وإلا انتقل إلى بكسل جديد.

ت- حساب أصغر دليل من بين أدلة البكسلات التي تنتمي إلى اليد.

ث- كتابة ترتيب البكسل في مصفوفة.

بعد الانتهاء من هذه الخطوات، قم بطرح قيمة الدليل الأصغر التي تم الحصول عليها من المصفوفة الناتجة.

2- الحصول على المسقط العلوي لليد في الصورة، من خلال تنفيذ العمليات السابقة على الأعمدة من الأول حتى الأخير.

3- الحصول على المسقط اليميني لليد في الصورة، من خلال تنفيذ العمليات السابقة على الأسطر من الأخير حتى الأول.

إن كل خطوة من الخطوات الثلاث السابقة يمكن أن تنفذ بشكل مستقل عن الخطوتين الأخريين، وفي كل خطوة لدينا إجرائيتين تنفذان على التسلسل، هما:

- الأولى: العمليات الأربع، والتي يمكن تنفيذها بشكل أنبوبي.
- الثانية: عملية الطرح (ضبط المصفوفة).

وبذلك نحصل على تسريع ناتج عن استخدام التقنية الأنبوبية في كل خطوة، وتسريع ناتج عن تنفيذ الخطوات الثلاث على التفرع، وسندعو الدارة التي تنفذ كل خطوة بالمسلك (Thread) [17][18][19][20].

10- المخطط الصندوقي (Block Diagram):

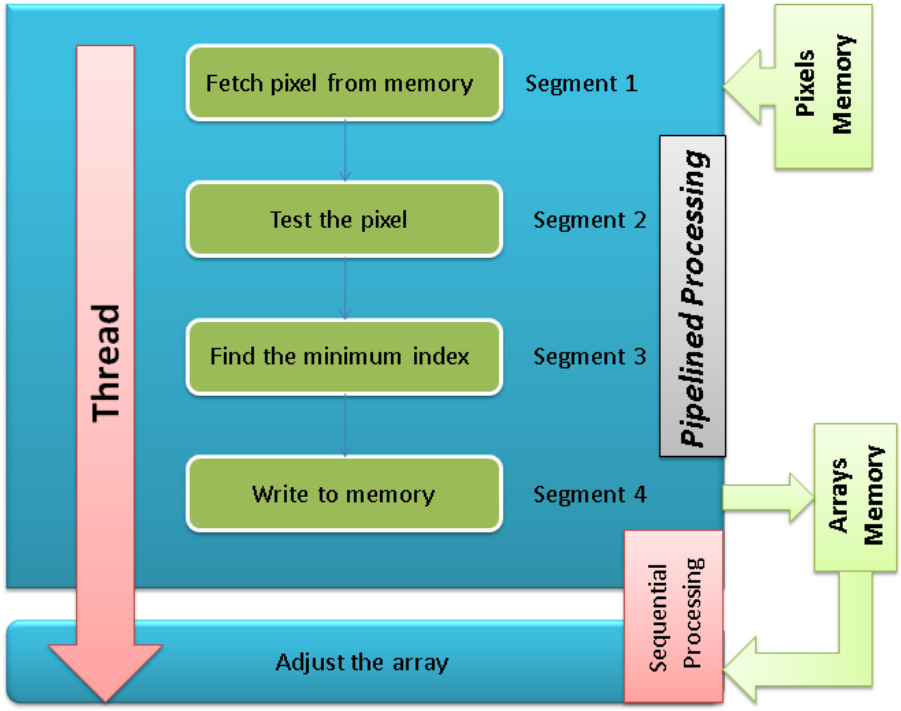
يبين الشكل (2) المخطط المبدئي لكل مسلك، إذ تتكون العمليات الأنبوبية فيه من أربعة مقاطع، وهي:

1- جلب بكسل من الذاكرة (Fetch pixel from memory).

2- اختبار البكسل (Test the pixel).

3- إيجاد العدد "الدليل" الأصغر (Find the minimum index).

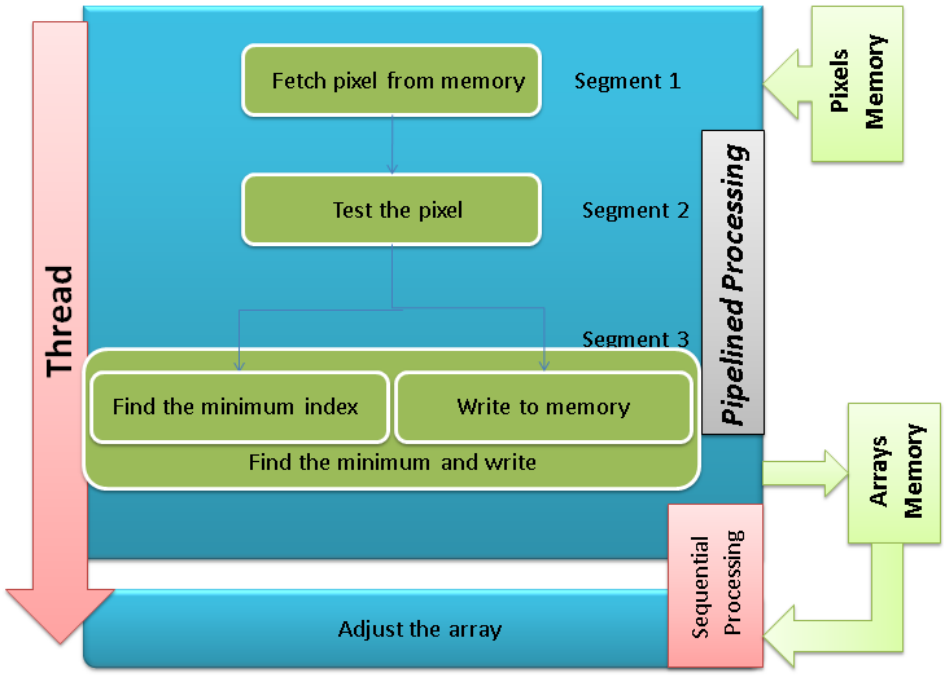
4- اكتب الترتيب في الذاكرة (Write to memory).



الشكل (2): المخطط المبدئي للمسلك.

بما أنه بالإمكان إجراء عملية الكتابة وتحديد العدد الأصغر بالوقت ذاته، فإنه يمكن دمج المرحلتين الأخيرتين في مرحلة واحدة هي مرحلة تحديد الأصغر والكتابة (Find the minimum and write).

يوضح الشكل (3) المسلك بعد دمج المرحلتين 3 و 4 في مرحلة واحدة.



الشكل (3): المخطط النهائي للمسلك بعد دمج المرحلتين 3 و 4.

سنقوم فيمايلي بشرح المقاطع الأربعة السابقة من أجل المسلك الخاص بأخذ مسقط ما. لكن قبل البدء بشرح مقاطع الأنبوب والمسالك، لا بد من الإشارة إلى أننا في هذا البحث سنعتمد على مصفوفة نعتبرها مصفوفة ثابتة معرفة كثابت ضمن شيفرة VHDL نسميها (a) تعبر عن الصورة، وكل خانة من خانات هذه المصفوفة مكونة من ثلاثة أرقام تعبر عن المركبات اللونية الثلاثة للبكسل الواحد، وهذه المصفوفة مبينة في الجدول (1).

	0	1	2	3	4
0	1,1,1(0)	2,2,2(1)	3,3,3(2)	4,4,4(3)	5,5,5(4)
1	12,10,14(5)	75,160,17(6)	15,80,158(7)	0,254,30(8)	0,5,8(9)
2	1,20,5(10)	6,6,6(11)	150,160,80(12)	120,200,250(13)	7,7,7(14)
3	8,8,8(15)	8,8,8(16)	9,9,9(17)	10,10,10(18)	250,0,0(19)
4	250,10,10(20)	150,10,10(21)	11,11,11(22)	12,12,12(23)	13,13,13(24)

الجدول (1): مصفوفة الصورة (a).

كما نرى، فإن كل عنصر في هذه المصفوفة مكون من الآتي:

رقم بين قوسين يمثل العنوان، أو الدليل للبكسل ضمن هذه المصفوفة.

ثلاثة أرقام تمثل البكسل بمركباته اللونية الثلاث، والتي هي الأحمر (r) والأخضر (g) والأزرق (b)، وكل منها يتدرج بين القيمتين 0 و 255.

الأرقام من 0 حتى 4 الموجودة أعلى المصفوفة تمثل ترتيب البكسل في العمود الذي هو فيه، والأرقام على اليسار تمثل ترتيب العنصر في السطر الذي هو فيه.

سنعتمد على تلك الأرقام في تشكيل المصفوفات المطلوبة.

نعتبر البكسل منتمياً إلى الخلفية "السوداء" إذا كان قريباً من اللون الأسود، أي إذا كانت مركباته أقل من 61 تماماً، ونعبر عن ذلك بالشرط المنطقي الآتي:

$$r<61 \text{ and } g<61 \text{ and } b<61$$

وإذا كان أي من تلك المركبات أكبر تماماً من 60، فإن البكسل يعتبر منتمياً إلى اليد، ونعبر عن ذلك بالشرط المنطقي الآتي:

$$r>60 \text{ or } g>60 \text{ or } b>60$$

استناداً لما سبق، فإن البكسلات التي تنتمي لليد تكون مظللة في المصفوفة السابقة، والبكسلات الأخرى تعتبر من الخلفية ولا تمثل اليد.

إذا قمنا بتطبيق الخوارزمية السابقة الذكر، نجد أن المصفوفات الناتجة هي على الشكل الآتي:

1. مصفوفة المسقط اليساري.
2. مصفوفة المسقط العلوي.
3. مصفوفة المسقط اليميني.

1- لتشكيل المصفوفة الأولى، نأخذ السطر الأول ابتداءً من اليسار (أي من البكسل (1,1,1) ذو الترتيب (0) في سطره)، فإذا كان هذا البكسل من اليد يتم وضع ترتيب هذا البكسل في سطره كعنصر في هذه المصفوفة، والانتقال إلى السطر التالي، وبعد الانتهاء من الأسطر جميعها يتم حساب العدد الأصغر في هذه المصفوفة، وطرحه من جميع عناصر هذه المصفوفة.

2- يتم تشكيل مصفوفة المسقط العلوي بالطريقة ذاتها، إذ يتم أخذ العمود الأول (الذي ترتيب بكسل ما في عموده هو (0))، وفحص البكسلات فيه، ولدى الوصول إلى بكسل ما من اليد يتم وضع ترتيب ذلك البكسل في عموده كعنصر في مصفوفة المسقط العلوي.

بعد ذلك، يتم حساب العنصر الأصغر في هذه المصفوفة وطرحه من المصفوفة الناتجة.

3- نحصل على مصفوفة المسقط اليميني، بتفحص عناصر السطر ابتداءً من اليمين، وتكرار الخطوات المذكورة بالنسبة للمسقط اليساري.

1.10- مقطع جلب بكسل من الذاكرة (Fetch Pixel Segment):

يعتبر هذا المقطع من أعقد المقاطع، لأنه يعتمد في عمله على محتويات مسجلات عدة في المقاطع الأخرى. وتتلخص وظيفة هذا المقطع في جلب بكسل ذي عنوان محدد من الذاكرة، حيث يتم تحديد هذا العنوان بالاعتماد على محتويات المسجلات في هذا المقطع ومقاطع أخرى.

سنذكر الآن المسجلات المشتركة لهذا المقطع لدى جميع المسالك، إذ ستكون هناك مسجلات أخرى إضافية تميز المسلك المعين، ويحتوي هذا المقطع على المسجلات الآتية:

- العداد pc: يتضمن هذا المسجل جزءاً من العنوان المستخدم في جلب البكسل من الذاكرة (في حالتنا هذه، المصفوفة (a)). بصيغة أوضح، فإن محتوى هذا العداد هو عدد يمثل ترتيب البكسل في الصف (بالنسبة لمصفوفتي المسقط اليميني واليساري)، وترتيب البكسل في العمود (بالنسبة لمصفوفة المسقط العلوي).

- العداد المساعد pline: يتضمن هذا المسجل قيمة تميز السطر الحالي أو العمود الحالي الذي يتم العمل به، وإذا تمت إضافة هذه القيمة إلى قيمة المسجل pc سنحصل على الدليل، أو العنوان الذاكري المحدد الذي سيمكننا من الوصول إلى البكسل المطلوب.

- المسجلات r, g, b: تتضمن هذه المسجلات القيم التي تعبر عن المركبات اللونية الثلاث التي تكون البكسل المعطى، إذ يحتوي المسجل r على مركبة اللون الأحمر والمسجل g على مركبة اللون الأخضر والمسجل b على مركبة اللون الأزرق، ويتم شحن هذه القيم إلى هذه المسجلات من الذاكرة مباشرة.

بالإضافة إلى تلك المسجلات، يحوي هذا المقطع وحدة التحكم بالعنوان (address control unit)، والتي تتحكم بمحتوى تلك المسجلات، كالقيام بعمليات

التصفير والزيادة والشحن بقيمة معينة. تعتمد هذه الوحدة في قرارها على محتوى العلام F من القطعة test pixel، بالإضافة إلى محتوى المسجلات التي تتحكم بها، وتعتبر هذه الوحدة الأعقد في الدارة كاملة، وهي التي تعطي هذه القطعة تعقيدها وتميزها من مسلك لآخر.

2.10- مقطع اختبار البكسل (Test Pixel Segment):

يتم في هذا المقطع معالجة القيم الموجودة في المسجلات r, g, b، والتأكد فيما إذا كان البكسل الذي تمثله هذه القيم منتمياً إلى اليد أم إلى الخلفية. وسنوضح بشكل أكبر وظيفة هذا المقطع بعد التعرف على مسجلاته الآتية:

- المسجل pctemp: يحمل هذا المسجل بالقيمة الموجودة في pc عند كل نبضة ساعة، وتكون وظيفته هي الاحتفاظ مؤقتاً بتلك القيمة.
- المسجل index1: يتم شحن هذا المسجل بالقيمة الموجودة في المسجل pctemp. وبذلك، فإن هذا المسجل سيحتوي على قيمة pc للبكسل الذي يسبق البكسل الحالي الذي تتم معالجته في هذه القطعة.
- العلام F: لهذا العلام أهميته البالغة، إذ يتم التحكم به من قبل وحدة فحص البكسل، فإذا كان البكسل الحالي منتمياً إلى اليد سيتم شحن العلام بالقيمة المنطقية '1'، وإذا كان منتمياً إلى الخلفية سيتم شحنه بالقيمة المنطقية '0'.

تتضمن هذه القطعة دارة تركيبية تدعى وحدة اختبار البكسل (Pixel Check Unit)، والتي تقوم بمقارنة محتوى المسجلات r, g, b مع القيمة 61 والتحقق من الشرط الآتي:

$$r<61 \text{ and } g<61 \text{ and } b<61$$

فإذا كان هذا الشرط محققاً، يتم شحن العلام F بالقيمة المنطقية '0'، وهذا يوافق حالة انتماء البكسل إلى الخلفية. وفي حالة عدم تحقق الشرط، سيتم شحن العلام بالقيمة المنطقية '1'، للدلالة على أن هذا البكسل ينتمي لليد.

3.10- مقطع إيجاد الدليل الأصغر والكتابة (Find the Minimum and Write):

يعتمد هذا المقطع في عمله بشكل جزئي على محتوى العلام F. فإذا كان محتواه

يشير إلى أن البكسل من اليد، فإنه ستتم مقارنة محتوى المسجل index1 مع القيمة الصغرى المفترضة، والموجودة في المسجل themin. وفي الوقت ذاته، كتابة محتوى المسجل index1 ضمن مصفوفة المسقط في الموقع الذي يحدده المسجل pcwr. وبالتالي، فإن هذه القطعة تتضمن المسجلات الآتية:

- المسجل themin: يحوي القيمة الأصغر لمصفوفة المسقط الناتجة قبل ضبطها في الجزء الثاني من المسلك، ويتم شحن هذا المسجل بقيمة بدائية 4.
- المسجل pcwr: يحتوي على العنوان، أو الترتيب الذي سيتم فيه تخزين قيمة المسجل index1، عندما تكون قيمة العلام F موافقة في مصفوفة المسقط، وتتم زيادة هذا المسجل بمقدار واحد بعد كل عملية كتابة.

يحتوي هذا المقطع على وحدة مقارنة من أجل مقارنة قيمة محتوى المسجل index1 مع محتوى المسجل themin. فإذا كان index1<themin، سيتم تفعيل خرج هذه الوحدة ليتم شحن قيمة محتوى المسجل index1 إلى المسجل themin، وإلا سيبقى محتوى themin بدون تغيير. وبذلك، سنحصل على القيمة الصغرى لمصفوفة المسقط الناتجة في نهاية عمل هذا الجزء، ليتم استخدامها في الجزء التالي.

11- المخطط الوظيفي (Functional Scheme):

يبين المخطط الموضح في الشكل (4) القطع الثلاث مع المسجلات والوحدات ضمن كل قطعة. ولا بد من الإشارة في هذا المخطط إلى أن نبضة الساعة (clk) موصولة إلى جميع المسجلات، ولم نقم بوصلها على الشكل من أجل تجنب التعقيد عليه.

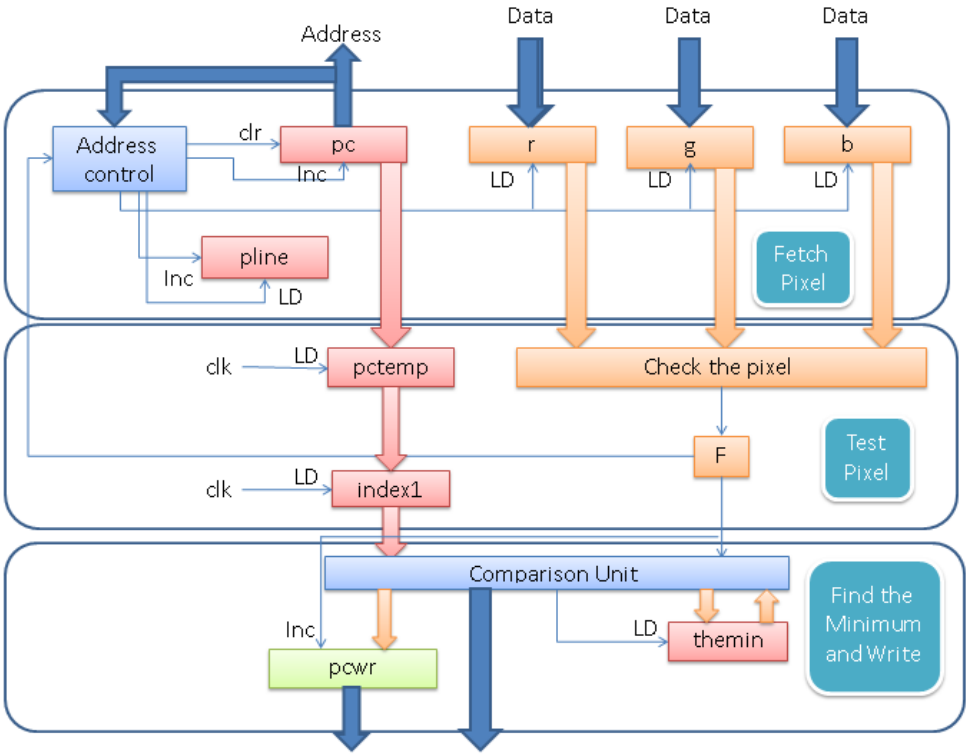
يتم تنفيذ هذا الشكل على شريحة FPGA باستخدام برنامج VHDL، إذ يحتوي هذا البرنامج على الترميز المطلوب من أجل تنفيذ المسالك الثلاثة، وكل من هذه المسالك يحوي جزأين (Two blocks): الجزء الأول مكون من ثلاث قطع أنبوبية، والجزء الثاني يتم تنفيذه بعد الانتهاء من الجزء الأول.

12- التنفيذ (Execution):

1.12- النتائج ومناقشتها (Results Discussion):

تطبيقاً لآلية عمل جميع العناصر والمخططات الوظيفية ولخوارزمية المعالجة

المطورة وبالاعتماد على البنية المصممة الموضحة في الشكل (3) والشكل (4) وباستخدام المصفوفة (a) المبينة في الجدول (1) على أنها مصفوفة الصورة المطلوب معالجتها ولجميع ما تم تقديمه وتعريفه وتوضيحه وشرحه سابقاً ضمن هذا البحث، فإن عمليات معالجة المساقط الثلاثة تتم وفق الخطوات I & II & III. ولكن للتوضيح أكثر، سنشرح آلية عمل النظام المصمم وطريقة تحصيل النتائج من أجل المسقط اليساري بالتزامن مع دورات الساعة، والتي تبرهن نجاح هذا النظام والخوارزمية المطورة في اكتشاف البكسلات المنتمية إلى اليد، وبالنتيجة أن النظام عملياتي، وهي الآلية المتبعة في تحصيل جميع النتائج.



الشكل (4): المخطط الوظيفي.

I– يبين الجدول (2) محتوى المسجلات عند كل نبضة ساعة من أجل جميع مسجلات الجزء الأول من المسلك الأول، أي مسلك المسقط اليساري، كما يوضح الشكل (5) نتائج المحاكاة.

الدورة الأولى: cycle=1

pc=1: عنوان العنصر الأول في السطر (0).

pline=0: عملية المعالجة تتم في السطر (0).

r,g,b=1,1,1: العنصر الأول في السطر (0) من المصفوفة (a).

endline=0: لا يمثل العنصر الحالي نهاية السطر (0).

pctemp=0: يحوي قيمة pc السابقة.

Index1=0: يحوي قيمة pc للبكسل الذي يسبق البكسل الحالي، والذي تتم معالجته في المقطع الحالي.

F=0: البكسل الحالي ينتمي إلى الخلفية، أي لا ينتمي إلى اليد.

themin=4: يحوي القيمة الأصغر لمصفوفة المسقط، ويشحن بداية بالقيمة 4.

pcwr=0: يحتوي على العنوان الذي سيخزن فيه دليل البكسل عندما يكون منتمياً إلى اليد، ولكن بما أن F=0 فالبكسل الحالي لا ينتمي إلى اليد بل إلى الخلفية.

q=0: خرج مقطع إيجاد الدليل الأصغر والكتابة، وتتغير قيمته حسب قيمة F.

finish1=0: نهاية المقطع الحالي، وهو حالياً المسقط اليساري الذي لم ننتهي منه بعد.

وهكذا، نستمر بعملية المعالجة إلى أن نصل إلى نهاية السطر (0)، فيكون لدينا التغيرات الآتية:

cycle=5: الدورة الخامسة

pc=0: وصلنا إلى البكسل الأخير بنهاية السطر (0).

r,g,b=5,5,5: العنصر الأخير بنهاية السطر (0).

endline=1: نهاية السطر (0).

pctemp=4: نفس الشرح أعلاه.

Index1=3: نفس الشرح أعلاه.

عندما نبدأ بمعالجة العنصر الأول (r,g,b=12,10,14) في السطر (1)، يكون لدينا التغيرات الآتية:

cycle=6: الدورة السادسة

pc=1: عنوان العنصر الأول في السطر (1).

pline=5: يمثل العنصر الذي ترتيبه (5)، وهو (r,g,b=12,10,14)، وكذلك يكون ترتيب باقي العناصر التي تقع في السطر (1) ذاته، مثل العنصر (r,g,b=75,160,17) والعنصر (r,g,b=15,80,158).

r,g,b=12,10,14: العنصر الأول في السطر (1).

وهكذا، تستمر عملية المعالجة إلى أن يتم اكتشاف بكسل ينتمي إلى اليد، فيكون لدينا التغيرات الآتية:

cycle=7: الدورة السابعة

pc=2: عنوان العنصر الثاني في السطر (1).

pline=5: عملية المعالجة تتم في السطر (1).

r,g,b=75,160,17: العنصر الثاني في السطر (1)، وهو يحقق شرط الانتماء إلى اليد (r or g or b)>60، حيث لدينا (r=75)>60 و (g=160)>60.

F=1: البكسل المكتشف ينتمي إلى اليد.

وكذلك F=1 في الدورة الثامنة (cycle=8)، وبالتالي في الدورة التالية للساعة يكون لدينا التغيرات الآتية:

themin=1: يشير إلى أن مصفوفة المسقط بدأت تستقبل عناصراً تنتمي إلى اليد.

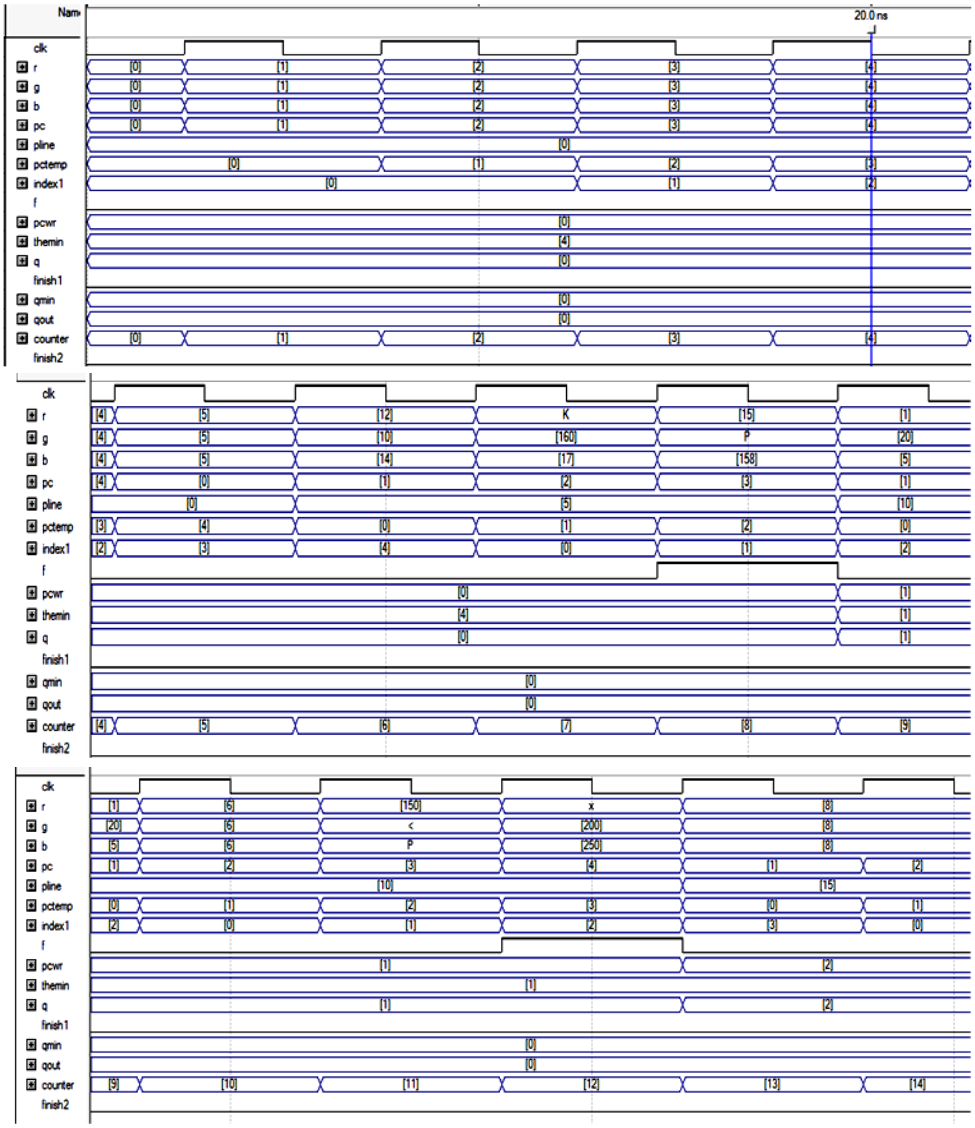
pcwr=1: يشير إلى العنوان الذي سيخزن فيه دليل العنصر المكتشف.

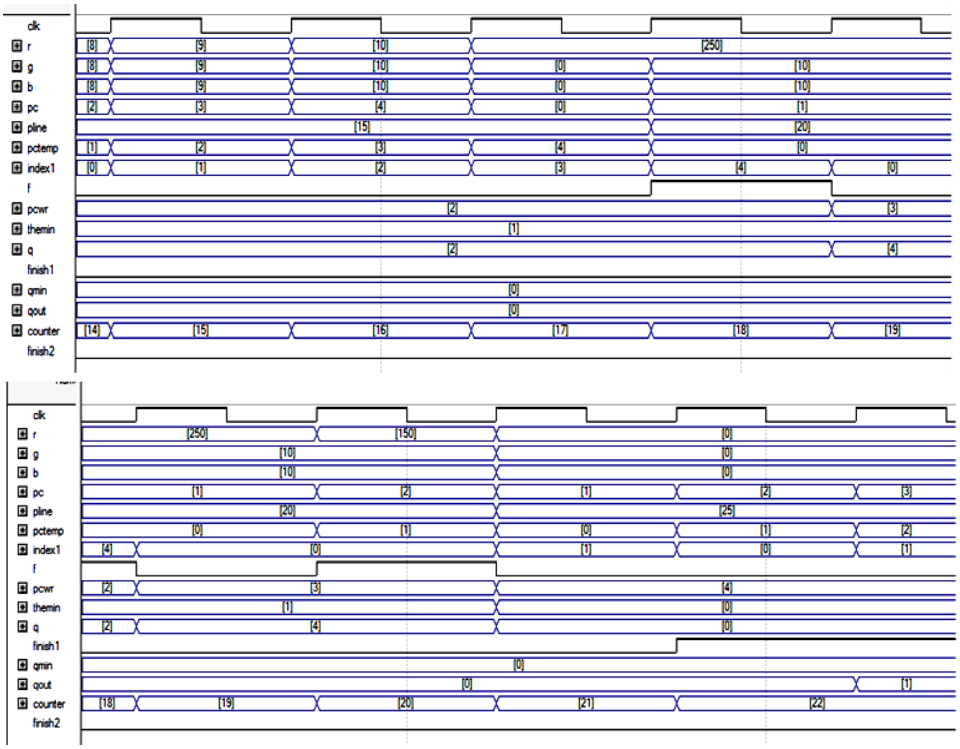
q=1: تفعيل خرج مقطع إيجاد الدليل الأصغر بعد أن تم اكتشاف العنصر.

وهكذا إلى أن نصل إلى نهاية عملية المحاكاة للمسقط اليساري، وعندها يكون finish1=1 حسب عدد القيم المدخلة.

Cycle	pc	pline	r,g,b	endline	pctemp	index1	F	themin	pcwr	q	finish1
1	1	0	1,1,1	0	0	0	0	4	0	0	0
2	2	0	2,2,2	0	1	0	0	4	0	0	0
3	3	0	3,3,3	0	2	1	0	4	0	0	0
4	4	0	4,4,4	0	3	2	0	4	0	0	0
5	0	0	5,5,5	1	4	3	0	4	0	0	0
6	1	5	12,10,14	0	0	4	0	4	0	0	0
7	2	5	75,160,17	0	1	0	1	4	0	0	0
8	3	5	15,80,158	0	2	1	1	4	0	0	0
9	1	10	1,20,5	0	0	2	0	1	1	1	0
10	2	10	6,6,6	0	1	0	0	1	1	1	0
11	3	10	150,60,80	0	2	1	1	1	1	1	0
12	4	10	120,200,250	0	3	2	1	1	1	1	0
13	1	15	8,8,8	0	4	3	0	1	2	2	0
14	2	15	8,8,8	0	1	4	0	1	2	2	0
15	3	15	9,9,9	0	2	1	0	1	2	2	0
16	4	15	10,10,10	0	3	2	0	1	2	2	0
17	0	15	250,0,0	1	4	3	1	1	2	2	0
18	1	20	250,10,10	0	0	4	1	1	2	2	0
19	1	20	250,10,10	0	0	0	1	1	3	4	0
20	2	25	150,10,10	0	1	0	1	1	3	4	0
21	1	25	11,11,11	0	0	1	0	0	4	0	0
22	2	25	0,0,0	0	1	0	0	0	4	0	1

الجدول (2): محتوى المسجلات في الجزء الأول من المسلك الأول خلال المعالجة.



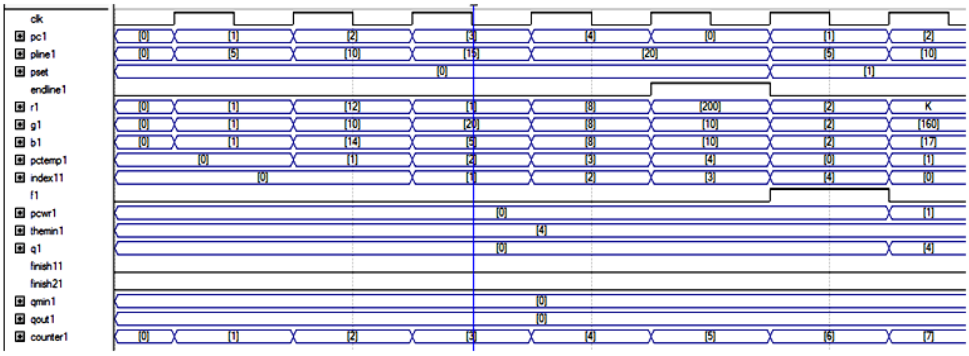


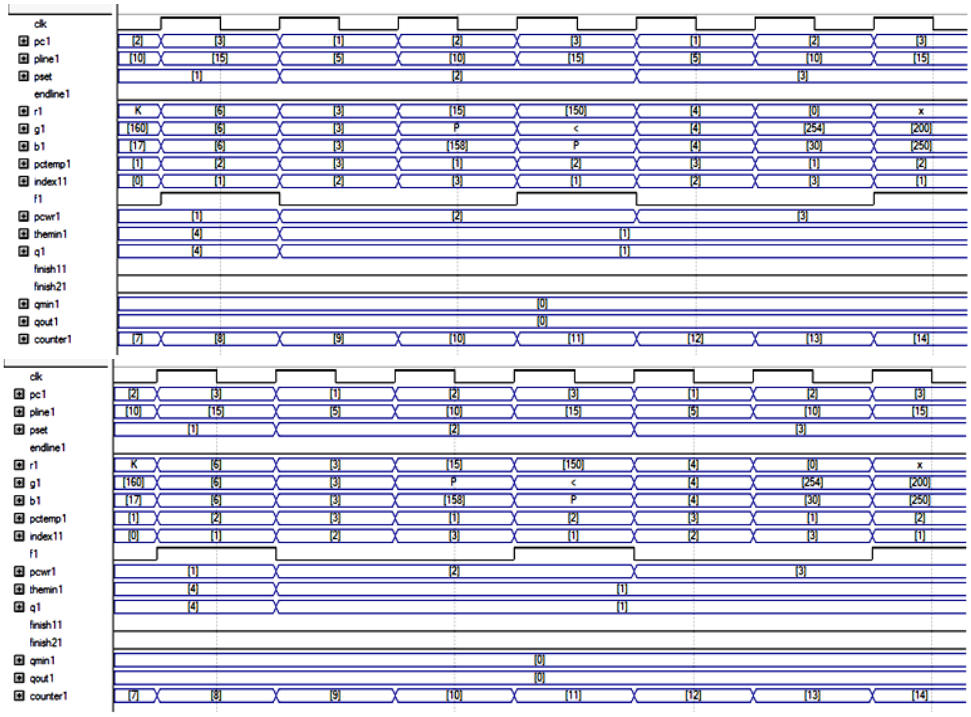
الشكل (5): نتائج المحاكاة للجزء الأول من المسلك الأول.

II– يبين الجدول (3) محتوى المسجلات عند كل نبضة ساعة من أجل جميع مسجلات الجزء الأول من المسلك الثاني، أي مسلك المسقط العلوي، كما يوضح الشكل (6) نتائج المحاكاة. وتتم عملية المحاكاة واستنتاج الجدول بطريقة مشابهة تماماً للمسقط اليساري الذي شرحناه أعلاه.

cycle	pc1	pline1	pset	end-line1	r,g,b	pct-emp	index1	f1	pcwr1	themin1	q1	finish11
1	1	5	0	0	1,1,1	0	0	0	0	4	0	0
2	2	10	0	0	12,10,14	1	0	0	0	4	0	0
3	3	15	0	0	1,20,5	2	1	0	0	4	0	0
4	4	20	0	0	8,8,8	3	2	0	0	4	0	0
5	0	20	0	1	200,10,10	4	3	0	0	4	0	0
6	1	5	1	0	2,2,2	0	4	1	0	4	0	0
7	2	10	1	0	75,160,17	1	0	0	1	4	4	0
8	3	15	1	0	6,6,6	2	1	1	1	4	4	0
9	1	5	2	0	3,3,3	3	2	0	2	1	1	0
10	2	10	2	0	15,80,158	1	3	0	2	1	1	0
11	3	15	2	0	150,60,80	2	1	1	2	1	1	0
12	1	5	3	0	4,4,4	3	2	0	3	1	1	0
13	2	10	3	0	0,254,30	1	3	0	3	1	1	0
14	3	15	3	0	120,200,250	2	1	1	3	1	1	0
15	1	5	4	0	5,5,5	3	2	0	4	1	1	0
16	2	10	4	0	0,5,8	1	3	0	4	1	1	0
17	3	15	4	0	7,7,7	2	1	0	4	1	1	0
18	4	20	4	0	250,0,0	3	2	0	4	1	1	0
19	0	20	4	1	13,13,13	4	3	1	4	1	1	0
20	1	5	5	0	12,10,14	0	4	0	5	1	3	0
21	2	10	5	0	1,20,5	1	0	0	5	1	3	1

الجدول (3): محتوى المسجلات في الجزء الأول من المسلك الثاني خلال المعالجة.



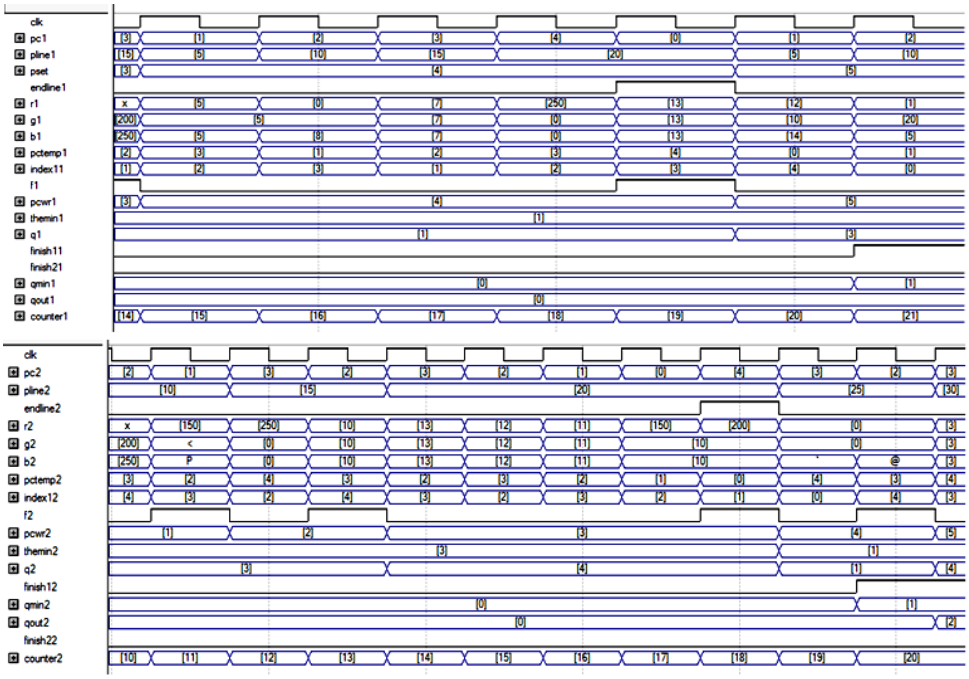


الشكل (6): نتائج المحاكاة للجزء الأول من المسلك الثاني.

III– يبين الجدول (4) محتوى المسجلات عند كل نبضة ساعة من أجل جميع مسجلات الجزء الأول من المسلك الثالث، أي مسلك المسقط اليميني، كما يوضح الشكل (7) نتائج المحاكاة. وتتم عملية المحاكاة واستنتاج الجدول بطريقة مشابهة تماماً للمسقط اليساري الذي شرحناه أعلاه.

cycle	pc2	pline2	endline2	r,g,b	pctemp2	index12	f2	pc-wr2	the-min2	q2	finish2
1	3	0	0	5,5,5	4	4	0	0	4	0	0
2	2	0	0	4,4,4	3	4	0	0	4	0	0
3	1	0	0	3,3,3	2	3	0	0	4	0	0
4	0	0	0	2,2,2	1	2	0	0	4	0	0
5	4	0	1	1,1,1	0	1	0	0	4	0	0
6	3	5	0	0,5,8	4	0	0	0	4	0	0
7	2	5	0	0,254,30	3	4	0	0	4	0	0
8	1	5	0	15,80,158	2	3	1	0	4	0	0
9	3	10	0	7,7,7	4	2	0	1	3	9	0
10	2	10	0	120,200,250	3	4	0	1	3	3	0
11	1	10	0	150,60,80	2	3	1	1	3	3	0
12	3	15	0	250,0,0	4	2	0	2	3	3	0
13	2	15	0	10,10,10	3	4	1	2	3	3	0
14	3	20	0	13,13,13	2	3	0	3	3	4	0
15	2	20	0	12,12,12	3	2	0	3	3	4	0
16	1	20	0	11,11,11	2	3	0	3	3	4	0
17	0	20	0	150,10,10	1	2	0	3	3	4	0
18	4	20	1	200,10,10	0	1	1	3	3	4	0
19	3	25	0	0,0,0	4	0	0	4	1	1	0
20	2	25	0	0,0,0	3	4	1	4	1	1	1

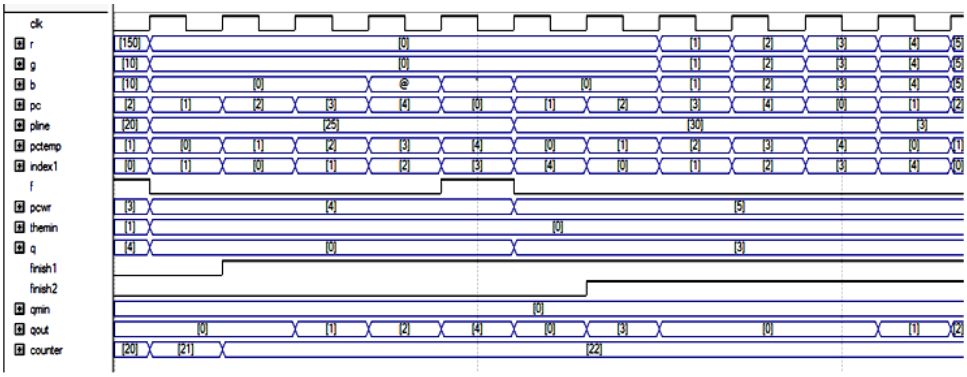
الجدول (4): محتوى المسجلات في الجزء الأول من المسلك الثالث خلال المعالجة.



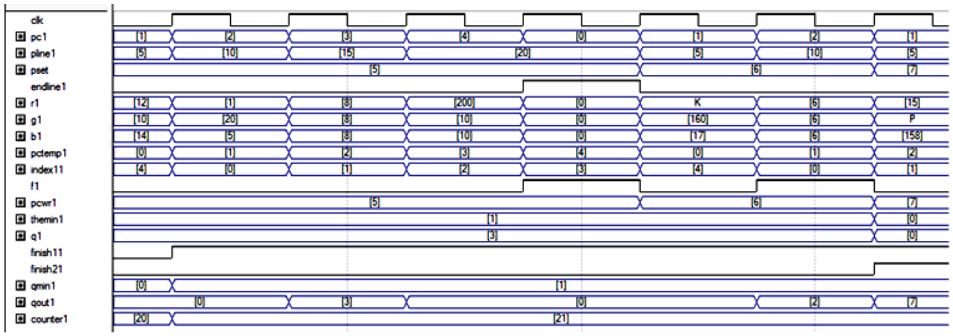
الشكل (7): نتائج المحاكاة للجزء الأول من المسلك الثالث.

بالنسبة للجزء الثاني، فإنه يتألف من دارة طارح ودارة جلب تقوم بإحضار عناصر مصفوفات المسقط، وتطرح منها قيمة العنصر الأصغر themin، وذلك باستخدام المسجلات q, qmin, qout. وتتم عملية المحاكاة بطريقة مشابهة تماماً للمسقط اليساري الذي شرحناه أعلاه.

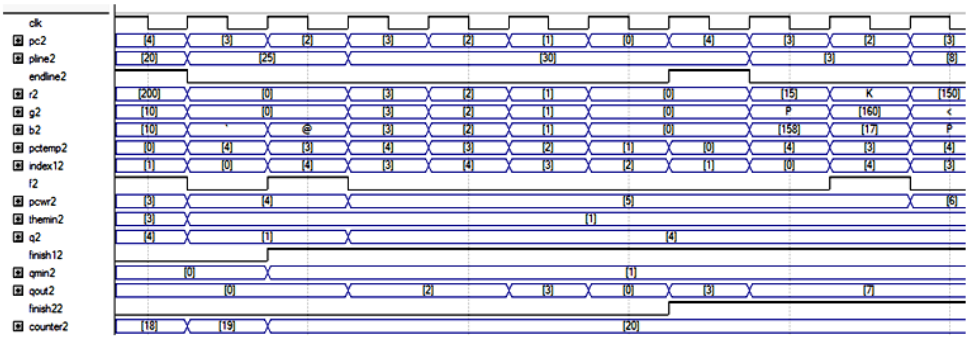
تبدأ هذه الدارة عملها عندما يتم تفعيل الإشارة finish1، وتنتهي عندما يتم تفعيل الإشارة finish2. وبالتالي، فإن النتيجة النهائية تكون بيانياً محصورة بين الإشارتين السابقتين في حالتهما المنطقية العليا، والشكل (8) يبين ذلك.



المصفوفة النهائية للمسقط اليساري.



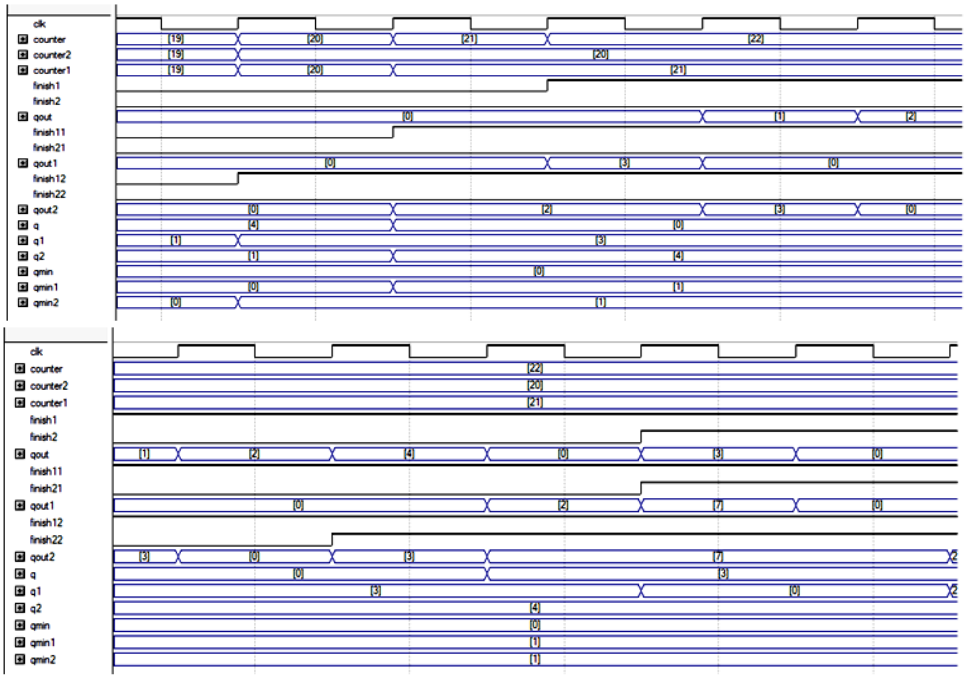
المصفوفة النهائية للمسقط العلوي.



المصفوفة النهائية للمسقط اليميني.

الشكل (8): المساقط بين إشارتي finish1 و finish2.

لدى تطبيق البرنامج كاملاً، نحصل على المخطط الموضح في الشكل (9) الناتج عن عمل جميع المسالك على التوازي. وتتم عملية المحاكاة بطريقة مشابهة تماماً للمسقط اليساري الذي شرحناه أعلاه.



الشكل (9): نتائج البرنامج النهائي.

2.12- حساب التسريع (Speedup Calculation):

سنقوم الآن بحساب التسريع الحاصل نتيجة استخدام التقنية الأنبوبية والمسالك المتوازية، وذلك بالمقارنة مع الحالة التسلسلية التي يتم فيها معالجة القطع في الجزء الأول من كل مسلك بشكل تسلسلي (حسب الشكل (2) قبل دمج القطعتين 3 و 4)، وكذلك تتم معالجة المسالك بشكل تسلسلي.

1.2.12- الحالة التسلسلية (Sequential State):

إن عدد البكسلات التي تنتمي إلى الخلفية، والتي سيتم المرور عليها في حالة المسقط اليساري هو 12 بكسلاً، وعدد تلك التي تنتمي إلى اليد هو 4 بكسلات. باعتبار أن البكسل الذي ينتمي إلى الخلفية سيتم تجاوزه بعد المرحلة الثانية، أي سيستغرق الأمر

نبضتي ساعة، فإن عدد نبضات الساعة المطلوبة من أجل هذه البكسلات هو 24 نبضة ساعة. أما في حالة البكسلات التي تنتمي إلى اليد، فالأمر يتطلب 4 نبضات ساعة لكل بكسل، وبالتالي سنحتاج إلى 16 نبضة ساعة. ويكون المجموع الكلي لنبضات الساعة في الجزء الأول من المسلك هو 40 نبضة ساعة.

بعد الانتهاء من الجزء الأول، سنحتاج إلى نبضة ساعة واحدة للتحضير ونبضة ساعة واحدة لكل عملية طرح. وبالتالي، سنحتاج إلى 5 نبضات ساعة في هذه المرحلة.

» يكون مجموع نبضات الساعة المطلوبة من أجل حساب مصفوفة المسقط اليساري هو 45 نبضة.

» بتكرار هذه الحسابات من أجل المسقط اليميني، نجد أن عدد النبضات المطلوبة هو 39 نبضة.

» من أجل المسقط العلوي، يكون عدد النبضات 46 نبضة.

بالنتيجة، يكون المجموع الكلي للنبضات المطلوبة هو 130 نبضة.

2.2.12- الحالة التفرعية (Parallel State):

في هذه الحالة، يصبح لدينا ثلاث قطع أنبوبية تعمل معاً، وسيتطلب الأمر مرور كل بكسل على المراحل الثلاث.

» في حالة المسقط اليساري، يكون عدد البكسلات التي سيتم المرور عليها هو 19 بكسلاً. باعتبار أن البكسل الأول سيتطلب 3 نبضات ساعة، ثم معالجة كل بكسل خلال نبضة واحدة، والبكسل الأخير سيحتاج إلى نبضتين إضافيتين للانتهاء. وبالتالي، يصبح لدينا عدد النبضات هو 21=2+19 نبضة، بالإضافة إلى 5 نبضات من أجل ضبط المصفوفة. ويصبح لدينا المجموع الكلي 26 نبضة.

» في حالة المسقط اليميني، نجد أن عدد النبضات المطلوبة هو 25 نبضة.

» في حالة المسقط العلوي، سنحتاج إلى 24 نبضة.

مما سبق، نستنتج أن عملية المعالجة تنتهي بشكل كلي خلال 26 نبضة، ويكون التسريع الحاصل هو:

$$speedup = \frac{130}{26} = 25$$

اعتماداً على الميزات التي تتمتع بها FPGA بالمعالجة التفرعية في الزمن الحقيقي، وكذلك ميزات المعالجة التفرعية باستخدام تقنية المسالك والتقنية الأنبوبية، فقد عملنا على دمج جميع هذه الميزات ضمن تصميم واحد في النظام الذي قمنا بتطويره، لنحصل على نتائج جيدة من خلال نسبة التسريع الناتجة مقارنة بباقي الأنظمة والأبحاث الأخرى التي ذكرناها سابقاً، إذ يتمتع نظامنا بالمزايا والإيجابيات الآتية:

- زمن اتصال شبه معدوم أثناء عمليات توزيع المهام والمعالجة.
- عدم وجود تشاركية (استقلالية) في الموارد.
- قابلية إعادة التطوير والبرمجة على شكل كتل مستقلة.
- معالجة في الزمن الحقيقي.
- صغر الحجم.
- كلفة منخفضة.
- استهلاك طاقة قليل.
- نظام مدمج ومحمول.

13- الخاتمة والآفاق المستقبلية (Conclusion and Perspectives):

يعتبر هذا البحث تجريبياً من أجل التأكد بشكل عملي من إمكانية تطبيق مبادئ المعالجة التفرعية بأشكالها المختلفة على خوارزمية المساقط المستخدمة من أجل معالجة صورة اليد، والحصول على المميزات منها، وذلك من خلال إنجاز نظام معالجة لإيماءات الصم والبكم. وبالنتيجة، نجاح النظام التفرعي المطور من خلال تطبيقه واختباره باستخدام FPGA مقارنة بالنظام التسلسلي.

لذلك، يمكن تطويره في المراحل المستقبلية لاستخدامه في تطبيقات الهواتف المحمولة. كما يمكن استعمال الشبكات العصبونية، إذ نستطيع تدريب الشبكة العصبونية على الحاسوب، ونقل المصفوفة الخاصة بها إلى ذاكرة الهاتف ليتم العمل انطلاقاً منها. وبذلك، يصبح لدينا نظاماً متكاملاً يمكنه تلبية المتطلبات المرجوة منه بشكل عملي.

14- المراجع (References):

[1] محمد هالة، "بعض الأساليب المعرفية لدى التلاميذ الصم والبكم وعلاقتها بالتحصيل الدراسي واتخاذ القرار"، مجمع الدكتور سليم طالب، الجزائر، 2013.

[2] Oi Mean Foong, Tan Jung Low, Satrio Wibowo, "Hand Gesture Recognition: Sign to Voice System (S2V)", International Journal of Electrical, Computer, and Systems Engineering 3:4, 2009.

[3] Tin Hninn Maung, "Real-Time Hand Tracking and Gesture Recognition System Using Neural Networks", World Academy of Science, Engineering and Technology 50, 2009.

[4] Ming-Hsuan Yang, Narendra Ahuja, Mark Tabb, "Extraction of 2D Motion Trajectories and Its Application to Hand Gesture Recognition", IEEE transactions on pattern analysis and machine intelligence, vol. 24, no. 8, august 2002.

[5] عيسى مراد، "الكمبيوتر والصم"، دار الوفاء، 2008.

[6] WWW.Egypalace.com

[7] بان أحمد متراس، آلاء محمود محمد، "استخدام المقدر اللبي وأسلوب العنقدة بمعدل k للتعرف على إيماءة اليد"، مجلة الرافدين لعلوم الحاسوب والرياضيات، 2013.

[8] Shweta Dour, J.M. Kundargi, "Design of ANFIS for Recognition of Signal Hand and Two Hand Signs for Indian Sign Language", JAIS, USA, 2013.

[9] Sanjay Saxena, Neeraj Sharma, Shiru Sharma, "Image Processing Tasks using Parallel Computing in Multi core Architecture and its Applications in Medical Imaging", International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE), Vol. 2, Issue 4, April 2013.

[10] Rafal Petryniak, “Analysis of Efficiency of Parallel Computing in Image Processing Task”, Institute of Applied Informatics, Cracow of Technology, 2008.

[11] S. McBader, “An FPGA implementation of a flexible, parallel image processing architecture suitable for embedded vision systems”, IEEE, NeuriCam S.p.A, Trento, Italy, 2008.

[12] Jari Nurmi, “Processor Design: System-on-Chip Computing for ASICs and FPGAs”, 2007.

[13] Ahmed Amine Jerraya, Wayne Wolf, “Multiprocessor Systems-on-Chips”, 2005.

[14] Enoch O. Hwang, “Microprocessor Design Principles and Practices with VHDL”, 2004.

[15] عمار زقزوق، "تطبيق بنية حساس فوق صوتي ضمن نظام قابل للبرمجة ونمذجته من أجل تطبيقات الروبوت"، مجلة جامعة البعث للعلوم الهندسية، 2011.

[16] Rafael C. Gonzalez, Richard E. Woods, Steven L. Eddins, “Digital Image Processing Using Matlab by Rafael”, 2012.

[17] عمار زقزوق، "نظم المعالجات التفرعية والبنى المتقدمة للحاسوب"، مطبوعات جامعة البعث، 2014.

[18] Hesham El-Rewini, Mostafa Abd-El-Barr, “Advanced Computer Architecture and Parallel Processing”, 2005.

[19] Alain Cazes, Joëlle Delacroix, “Architecture des Machines et des Systèmes Informatiques”, 2005.

[20] عمار زقزوق، "دراسة العوامل المؤثرة بنسبة الإصابة في الذاكرة المخبئية ضمن نظام متعدد المعالجات"، مجلة جامعة تكريت للعلوم الهندسية في العراق، 2014.

Develop a Parallel Processing System to Recognize the Deaf and Mute Gestures and Apply it based on FPGA

Dr. Eng. Ammar Zakzouk

Associate Professor in Mechanical and Electrical Engineering Faculty

Albaath University

Abstract

In this paper, we processed an array which represents the human hand image to get the characteristics of this image. So, we used FPGA technique, and the processing operation is partitioned into three threads which is carried out in parallel. Each thread is carried out using the pipeline technique by partitioning thread into four segments. After that, we evaluated the speedup that we get in result of using the pipeline technique and the parallel threads. So, we have the possibility to design an embedded system integrated into chip (SoC), and using the mobile phones as integral devices support the software and hardware resources.

Key-words: Deaf and Mute Gestures – Images Processing – Patterns Recognition – Pipeline Processing – Threads – Parallel Processing – Programmable Circuits (FPGA) – Hardware Description Language (VHDL) – Embedded System – System on Chip (SoC) - Speed-up.