



VIETNAM NATIONAL UNIVERSITY, HO CHI MINH CITY
 UNIVERSITY OF SCIENCE, HO CHI MINH CITY
 FACULTY OF ELECTRONICS & TELECOMMUNICATIONS
 COMPUTER & EMBEDDED SYSTEM LABORATORY

Face Detection Specification

Summary:

Version: 1.0		Release Date: Month: <u>May</u> Year: <u>2012</u>	Total Size: <input checked="" type="checkbox"/> Reports <input checked="" type="checkbox"/> References <input checked="" type="checkbox"/> Simulation	<input checked="" type="checkbox"/> Images <input checked="" type="checkbox"/> Source Codes
Solution Type: <input checked="" type="checkbox"/> IP Core <input type="checkbox"/> Megafunction <input type="checkbox"/> SOPC Builder Component <input type="checkbox"/> Qsys Component <input type="checkbox"/> System Design		External Devices: <input checked="" type="checkbox"/> Lcd TRDB-LCM		
Supported Device(s)/ Speed (MHz)/ LEs:	<input type="checkbox"/> Stratix IVMHzLEs	
	<input type="checkbox"/> Stratix IIIMHzLEs	
	<input checked="" type="checkbox"/> Stratix IIMHzLEs	
	<input type="checkbox"/> Cyclone IIIMHzLEs	
	<input type="checkbox"/> Cyclone IIMHzLEs	
Description:		Face Detection Core (Verilog HDL).		

Contents

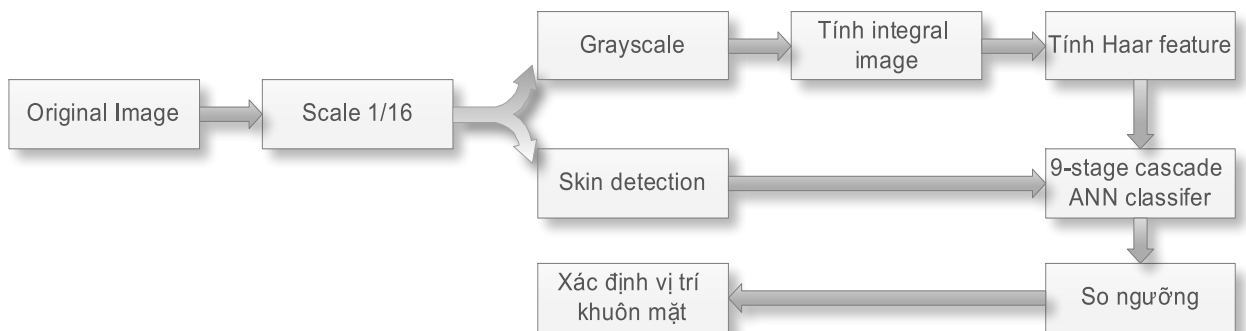
1	Quy trình nhận diện khuôn mặt:	5
2	Chi tiết thiết kế:	6
2.1	Pre_Processing_Core:	7
2.1.1	2D Modified Fast Haar Wavelet Transform:	7
2.1.1.1	Thuật toán 2D Modified Fast Haar Wavelet Transform:	7
2.1.1.2	Chi tiết thiết kế:	9
2.1.1.2.1	AVG:	9
2.1.1.2.2	PPBUFFER:	10
2.1.1.2.2.1	Sigle Clock FIFO:	10
2.1.1.2.2.2	SCFIFO_Nx4:	11
2.1.1.2.2.3	PPBUFFER_640x4:	12
2.1.1.2.2.4	BUFFER_160x4:	12
2.1.1.2.2.5	PPBUFFER:	13
2.1.1.2.3	CONTROL_UNIT:	14
2.1.2	Grayscale Convert:	15
2.1.2.1	Phép chuyển từ RGB sang YCbCr:	16
2.1.2.2	Chi tiết thiết kế:	16
2.1.2.2.1	r2Yr:	16
2.1.2.2.2	g2Yg:	16
2.1.2.2.3	b2Yb:	17
2.1.3	Skin Detector ANN:	18
2.1.3.1	Thuật toán Skin Detection ANN:	18
2.1.3.2	Chi tiết thiết kế;	20
2.1.3.2.1	MAC_R_GEN:	20
2.1.3.2.2	MAC_G_GEN:	21
2.1.3.2.3	MAC_B_GEN:	22
2.1.3.2.4	SUM_4WORD:	22
2.1.3.2.5	SIGMOID_APPROX:	23

2.1.3.2.6	MAC_GEN:	24
2.1.3.2.7	SUM_7WORD:	25
2.1.3.2.8	SIGMOID_APPROX_1:	25
2.1.3.2.9	CONTROL_UNIT:	26
2.1.4	Integral Image Generation:	27
2.1.4.1	Khái niệm và phương pháp tính Integral Image:	28
2.1.4.2	Chi tiết thiết kế:	29
2.1.4.2.1	MAC:	29
2.1.4.2.2	SUM:	30
2.1.4.2.3	SCFIFO_80x1:	31
2.1.4.2.4	PPBUFFER_80x1:	31
2.1.4.2.5	CONTROL_UNIT:	32
2.2	Detection_Core:	33
2.2.1	Haar Feature Generation:	33
2.2.1.1	Khái niệm và phương pháp tính các đặc trưng Haar:	34
2.2.1.2	Chi tiết thiết kế:	36
2.2.1.2.1	8WAYREC:	36
2.2.1.2.1.1	REC:	36
2.2.1.2.2	FEATURE COMPOSITION:	37
2.2.1.2.3	NORMALIZATION:	38
2.2.1.2.4	CONTROL UNIT:	39

Table of Figures

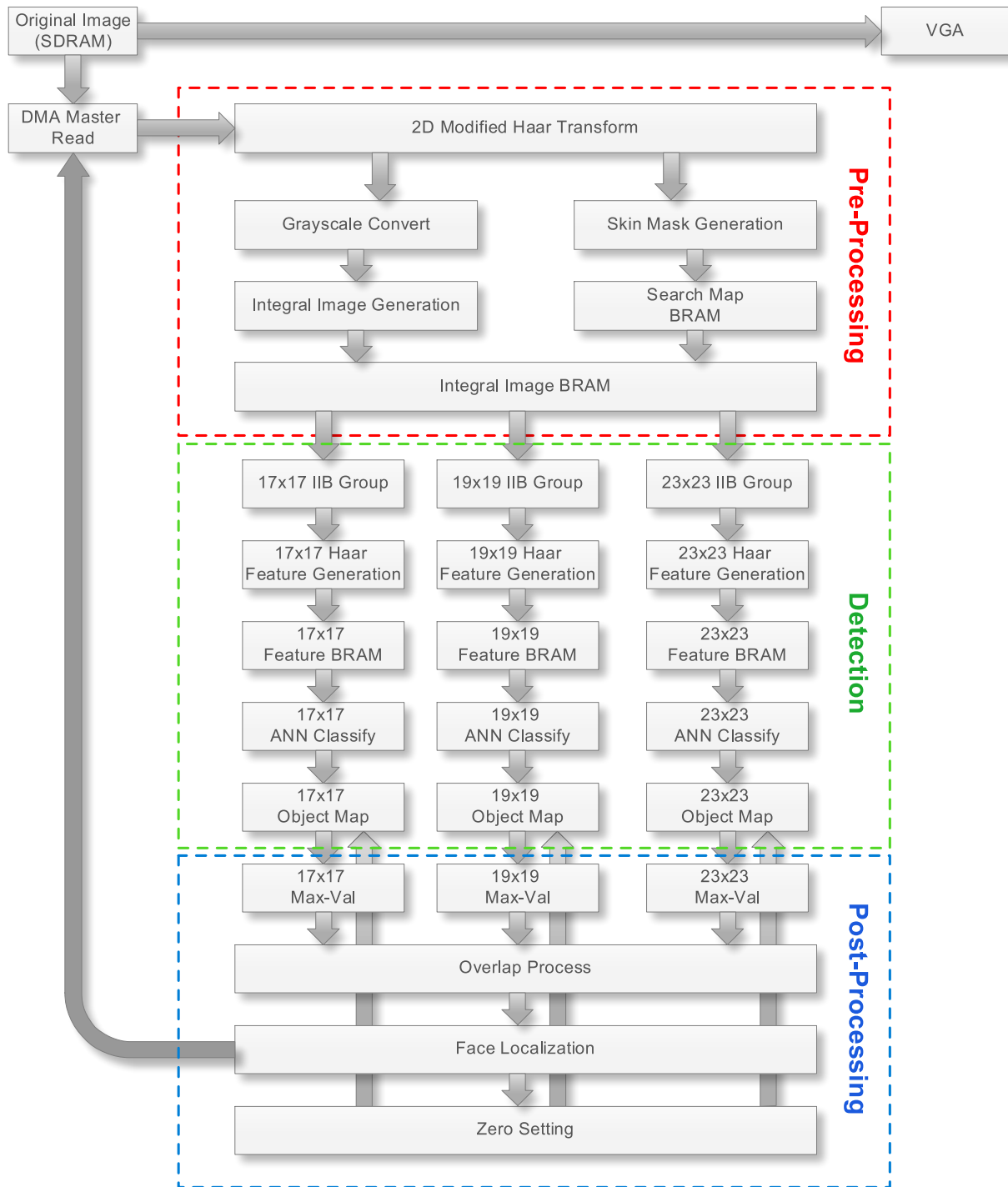
Hình 1: Quy trình thuật toán nhận diện khuôn mặt	5
Hình 2 : Sơ đồ khối nhận diện khuôn mặt	6
Hình 3: 2D Modified Fast Haar Transform	7
Hình 4: AVG trong Khối PRE-PROCESSING	9
Hình 5: SCFIFO trong Khối SCFIFO_Nx4	10
Hình 6: SCFIFO_Nx4 trong Khối PPBUFFER	11
Hình 7: PPBUFFER_640x4 trong Khối 2D MFHWT	12
Hình 8: BUFFER_160x4 trong Khối 2D MFHWT	13
Hình 9: PPBUFFER trong Khối 2D MFHWT	13
Hình 10: CONTROL_UNIT trong Khối 2D MFHWT	14
Hình 11: Grayscale Convert	15
Hình 12: r2Yr trong Khối Grayscale Convert	16
Hình 13: g2Yg trong Khối Grayscale Convert	17
Hình 14: b2Yb trong Khối Grayscale Convert	17
Hình 15: Skin Detector ANN trong Khối Pre_Processing	18
Hình 16: Skin Detector ANN	19
Hình 17: Đồ thị hàm Sigmoid Fuction	20
Hình 18: MAC_R_GEN trong Khối Skin Detector ANN	20
Hình 19: MAC_G_GEN trong Khối Skin Detector ANN	21
Hình 20: MAC_B_GEN trong Khối Skin Detector ANN	22
Hình 21: SUM_4SWORD trong Khối Skin Detector ANN	23
Hình 22: SIGMOID_APPROX trong Khối Skin Detector ANN	24
Hình 23: MAC_GEN trong Khối SKin Detector ANN	24
Hình 24: SUM_7WORD	25
Hình 25: SIGMOID_APPROX_1 trong Khối Skin Detector ANN	26
Hình 26: CONTROL_UNIT trong Khối Skin Detector ANN	27
Hình 27: IIG trong Khối Pre_Processing	28
Hình 28: Cấu trúc Integral Image	29
Hình 29: Phương pháp tính Integral Image	29
Hình 30: MAC trong Khối IIG	30
Hình 31: SUM trong Khối IIG	30
Hình 32: PPBUFFER_80x1 trong Khối IIG	31
Hình 33: CONTROL_UNIT trong Khối IIG	32

1 Quy trình nhận diện khuôn mặt:



Hình 1: Quy trình thuật toán nhận diện khuôn mặt

2 Chi tiết thiết kế:

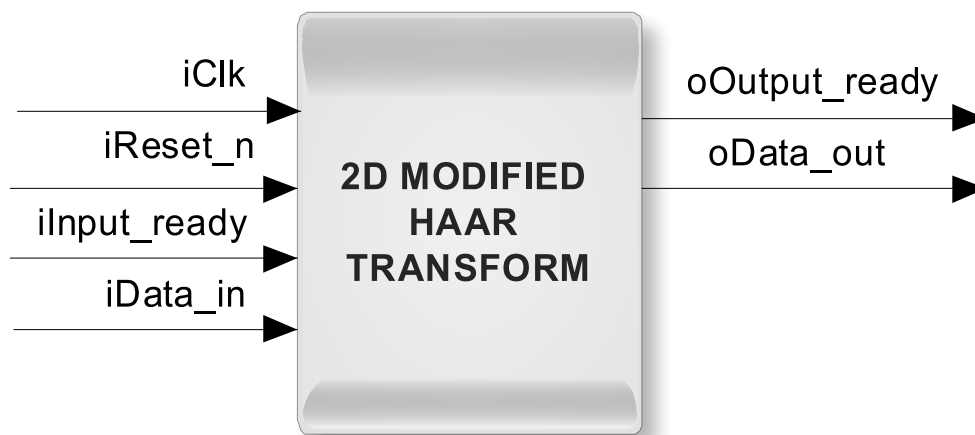


Hình 2 : Sơ đồ khối nhận diện khuôn mặt

2.1 Pre_Processing_Core:

2.1.1 2D Modified Fast Haar Wavelet Transform:

Nhiệm vụ của khối 2D MFHWT là scale ảnh RGB 1280x960 thành ảnh RGB 80x60.



Hình 3: 2D Modified Fast Haar Transform

Tín hiệu	Độ rộng	Ghi/ Đọc	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iReset_n	1	Input	Tín hiệu Reset đồng bộ tác động mức thấp.
iInput_ready	1	Input	Tín hiệu thông báo dữ liệu vào hợp lệ từ DMA Read.
iData_in	64	Input	Dữ liệu từ DMA Read vào Pre_Pocessing_Core
oOutput_ready	1	Output	Tín hiệu thông báo dữ liệu ra hợp lệ tới Grayscale Convert & Skin ANN Detector.
oData_out	16	Output	Dữ liệu sau khi scale 1/16.

2.1.1.1 Thuật toán 2D Modified Fast Haar Wavelet Transform:

Thuật toán 1D Modified Fast Haar Wavelet Transform : với 1 mảng gồm N phần tử $f = (f_1, f_2, f_3, \dots, f_N)$, ta lần lượt tính trung bình của 1 nhóm gồm 4 phần tử sẽ được 1 mảng N/4 phần tử $a = (a_1, a_2, a_3, \dots, a_{N/4})$:

$$a_m = \frac{f_{4m-3} + f_{4m-2} + f_{4m-1} + f_{4m}}{4}, m=1,2,3,\dots,N/4 \text{ (} a_m \text{: hệ số xấp xỉ)}$$

Tiếp theo, ta tính hệ số chi tiết của từng nhóm 4 phần tử đó sẽ được 1 mảng $N/2$ phần tử $d = (d_1, d_2, d_3, \dots, d_{N/2})$:

$$d_m = \begin{cases} a_m = \frac{(f_{4m-3} + f_{4m-2}) - (f_{4m-1} + f_{4m})}{4}, & m = 1, 2, 3, \dots, N/4 \\ 0, & m = N/2, \dots, N \end{cases}$$

Thuật toán 2D Modified Fast Haar Wavelet Transform : thuật toán 2D MFHWT trên ảnh ($M \times N$) được mở rộng từ 1D MFHWT. Các bước thực hiện :

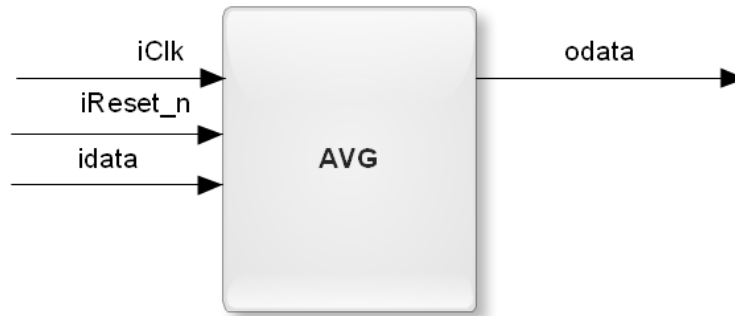
- Đọc ảnh $M \times N$ như 1 ma trận $M \times N$
- Áp dụng thuật toán 1D trên từng dòng (với từng mảng gồm N phần tử)
- Sau bước (b), ta lấy ma trận chuyển vị của ma trận vừa thu được
- Áp dụng lại thuật toán 1D trên từng dòng (với từng mảng gồm M phần tử)
- Lấy ma trận chuyển vị của ma trận vừa thu được , ta được ảnh đã được scale $1/4$ lần

Trong thiết kế, ta cần scale ảnh 1280×960 xuống ảnh 80×60 nên ta sẽ phải thực hiện 2 lần thuật toán 2D MFHWT (scale $1/16$). Do ta chỉ cần ảnh scale của ảnh gốc nên ta chỉ tính hệ số xấp xỉ mà không cần tính hệ số chi tiết. Để thiết kế hệ thống theo mô hình pipeline, ta sẽ cần có các bộ đệm (3 bộ) giữa các bộ tính trung bình (4 bộ). Trong thiết kế này, ta sẽ dùng FIFO, kích thước các bộ đệm này tùy từng giai đoạn:

- Tính trung bình lần 1 (trên dòng): ta không cần chờ cho hệ thống tính xong hết tất cả các dòng mà chỉ cần xong đủ 4 dòng là đủ dữ liệu cho bộ trung bình 2 sau xử lí. Do đó, bộ đệm 1 là 1 pingpong buffer gồm 2 buffer, mỗi buffer gồm 4 FIFO (320×2 bytes).
- Tính trung bình lần 2 (trên cột): do bộ trung bình 3 chỉ cần đủ 4 pixel là có thể xử lí nên ở đây ta chỉ cần dùng 2 register (64bits) thay cho 1 pingpong buffer.
- Tính trung bình lần 3 (trên dòng): thứ nhất, do bộ trung bình 4 cần đủ dữ liệu 4 dòng. Thứ hai, do cần 5440 xung clock để có đủ dữ liệu cho bộ trung bình 4 mà bộ 4 thì chỉ cần 80 xung clock để tính xong phần dữ liệu đó. Do đó, ta chỉ cần dùng 1 buffer gồm 4 FIFO (80×2 bytes).

2.1.1.2 Chi tiết thiết kế:

2.1.1.2.1 AVG:



Hình 4: AVG trong Khối PRE-PROCESSING

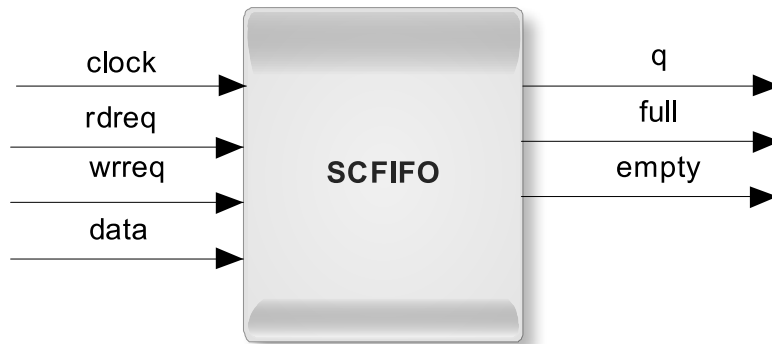
AVG là bộ tính trung bình của khối 2D MFHWT. Nhiệm vụ của bộ AVG: tách từng phần R,G,B của 4 pixel liên tiếp rồi tính trung bình từng thành phần sau đó kết hợp trở lại thành 1 pixel:

$$\begin{cases} R_o = (R_{i1} + R_{i2} + R_{i3} + R_{i4})/4 \\ G_o = (G_{i1} + G_{i2} + G_{i3} + G_{i4})/4 \\ B_o = (B_{i1} + B_{i2} + B_{i3} + B_{i4})/4 \end{cases} \rightarrow (R_o, G_o, B_o)$$

Tín hiệu	Độ rộng	Hướng	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iReset_n	1	Input	Tín hiệu Reset đồng bộ tác động mức thấp.
idata	64	Input	Dữ liệu từ DMA Read vào Pre_Pocessing_Core.
odata	16	Output	Dữ liệu sau khi tính trung bình tới PPBUFFER0.

2.1.1.2.2 PPBUFFER:

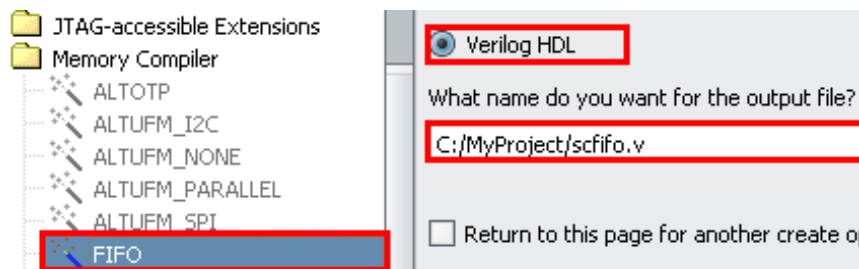
2.1.1.2.2.1 Single Clock FIFO:



Hình 5: SCFIFO trong Khối SCFIFO_Nx4

Phần mềm Quartus II của Altera hỗ trợ công cụ MegaWizard Plug-In Manager cho phép sử dụng các thiết kế có sẵn của Altera. Single Clock FIFO được lấy ra từ công cụ này. Quá trình chọn lựa và cấu hình thông số được thực hiện qua các bước sau:

- Trong Quartus II chọn Tools/MegaWizard Plug-In Manager...
- Chọn “Create a new custom megafunction variation” để tạo mới một thành phần nào đó.
- Chọn FIFO và đặt tên file verilog muốn xuất ra.



- Trong tab “Width, Clks, Synchronization”, lựa chọn các thông số về độ rộng đường dữ liệu, độ lớn của FIFO, và quan trọng là chọn chế độ single clock bằng cách chọn “Yes, synchronize both reading and writing to ‘clock’”.

How wide should the FIFO be? 8 bits

☐ Use a different output width and set to 8 bits

How deep should the FIFO be? 256 words

Note: You could enter arbitrary values for width and depth

Do you want a common clock for reading and writing the FIFO?

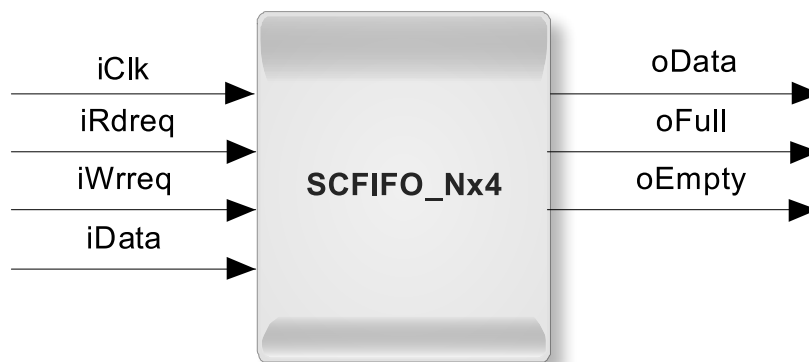
☒ Yes, synchronize both reading and writing to 'clock'.
Create one set of full/empty control signals.

- Lựa chọn các thông số khác cho phù hợp và sau đó Finish để hoàn tất quá trình tạo FIFO.

Tín hiệu	Độ rộng	Hướng	Mô tả
clock	1	Input	Tín hiệu xung nhịp đồng bộ dữ liệu
data	16	Input	Dữ liệu đưa vào FIFO.
rdreq	1	Input	Tín hiệu yêu cầu đọc dữ liệu từ FIFO.
wrreq	1	Input	Tín hiệu yêu cầu ghi dữ liệu tới FIFO.
full	1	Output	Tín hiệu báo FIFO đầy.
empty	1	Output	Tín hiệu báo FIFO rỗng.
q	16	Output	Dữ liệu lấy ra từ FIFO.

2.1.1.2.2.2 SCFIFO_Nx4:

SCFIFO_Nx4 là 1 buffer gồm 4 FIFO. N = 640 đối với bộ PPBUFFER_640x4 trong đó mỗi buffer gồm 4 FIFO (640bytes). N = 160 đối với bộ BUFFER_160x4 trong đó mỗi buffer gồm 4 FIFO (160bytes).



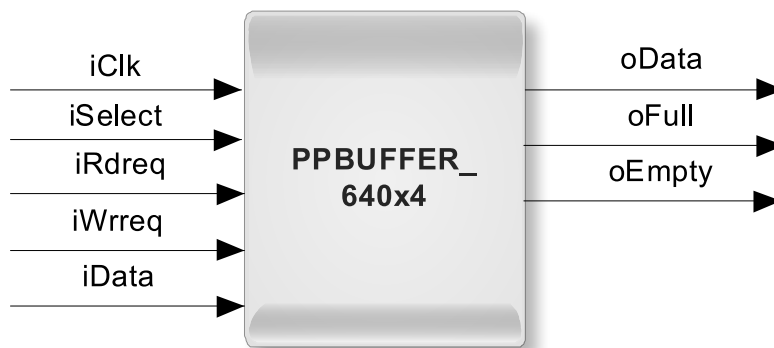
Hình 6: SCFIFO_Nx4 trong Khối PPBUFFER

Tín hiệu	Độ rộng	Hướng	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.

iData	16	Input	Dữ liệu đưa vào SCFIFO_Nx4.
iRdreq	1	Input	Tín hiệu yêu cầu đọc dữ liệu từ SCFIFO_Nx4.
iWrreq	4	Input	Tín hiệu yêu cầu ghi dữ liệu tới SCFIFO_Nx4.
oFull	4	Output	Tín hiệu báo trạng thái đầy của các FIFO bên trong.
oEmpty	1	Output	Tín hiệu báo tất cả các FIFO bên trong rỗng.
oData	64	Output	Dữ liệu lấy ra từ FIFO.

2.1.1.2.2.3 PPBUFFER_640x4:

PPBUFFER_640x4 là 1 pingpong buffer gồm 2 buffer SCFIFO_640x4. Đây là bộ đệm giữa AVG0 và AVG1.

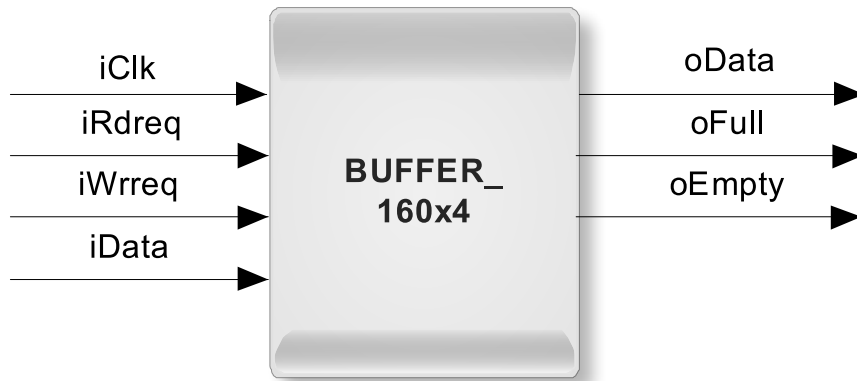


Hình 7: PPBUFFER_640x4 trong Khối 2D MFHWT

Tín hiệu	Độ rộng	Hướng	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iSelect	1	Input	Tín hiệu chọn buffer bên trong để đọc và ghi.
iData	16	Input	Dữ liệu đưa vào PPBUFFER_640x4 từ AVG.
iRdreq	2	Input	Tín hiệu yêu cầu đọc dữ liệu từ PPBUFFER_640x4.
iWrreq	8	Input	Tín hiệu yêu cầu ghi dữ liệu tới PPBUFFER_640x4.
oFull	8	Output	Tín hiệu báo trạng thái đầy của các FIFO bên trong.
oEmpty	2	Output	Tín hiệu báo tất cả các FIFO bên trong mỗi SCFIFO_640x4 rỗng.
oData	64	Output	Dữ liệu lấy ra từ PPBUFFER_640x4 tới AVG.

2.1.1.2.2.4 BUFFER_160x4:

BUFFER_160x4 là 1 buffer chỉ gồm 1 SCFIFO_160x4. Đây là bộ đệm giữa AVG1 và AVG2.

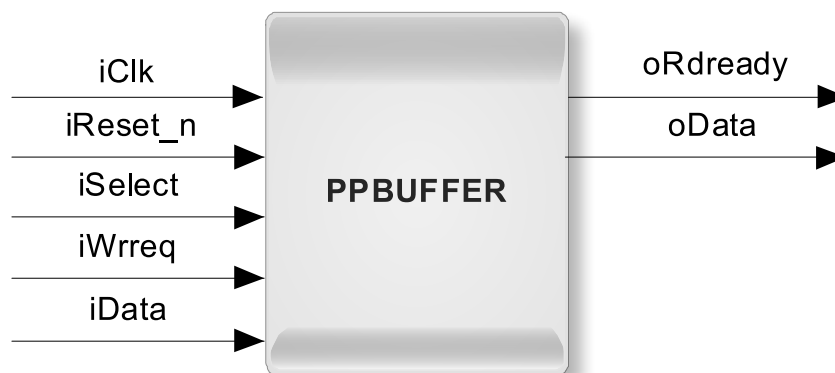


Hình 8: BUFFER_160x4 trong Khối 2D MFHWT

Tín hiệu	Độ rộng	Hướng	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iData	16	Input	Dữ liệu đưa vào BUFFER_160x4 từ AVG.
iRdreq	1	Input	Tín hiệu yêu cầu đọc dữ liệu từ BUFFER_160x4.
iWrreq	4	Input	Tín hiệu yêu cầu ghi dữ liệu tới BUFFER_160x4.
oFull	4	Output	Tín hiệu báo trạng thái đầy của các FIFO bên trong.
oEmpty	1	Output	Tín hiệu báo tất cả các FIFO bên trong SCFIFO_160x4 rỗng.
oData	64	Output	Dữ liệu lấy ra từ BUFFER_160x4 tới AVG.

2.1.1.2.2.5 PPBUFFER:

PPBUFFER là 1 buffer gồm 2 register 64bits. Đây là bộ đệm giữa AVG2 và AVG3.

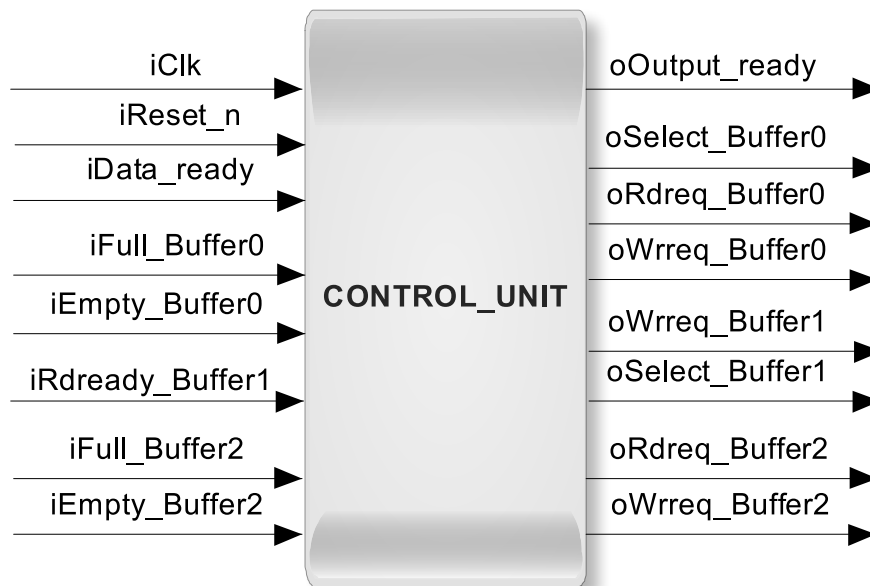


Hình 9: PPBUFFER trong Khối 2D MFHWT

Tín hiệu	Độ rộng	Hướng	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iReset_n	1	Input	Tín hiệu Reset đồng bộ tác động mức thấp.
iSelect	1	Input	Tín hiệu chọn register bên trong để đọc và ghi.
iWrreq	1	Input	Tín hiệu yêu cầu ghi dữ liệu tới PPBUFFER.
iData	16	Input	Dữ liệu đưa vào PPBUFFER từ AVG.
oRdready	1	Output	Tín hiệu báo dữ liệu sẵn sàng được đọc.
oData	64	Output	Dữ liệu lấy ra từ PPBUFFER tới AVG.

2.1.1.2.3 CONTROL_UNIT:

Nhiệm vụ của CONTROL_UNIT điều khiển cho khối 2D MFHWT làm việc một cách đồng bộ. Chủ yếu nhận các tín hiệu full và empty từ các bộ đệm để điều khiển việc đọc và ghi dữ liệu.



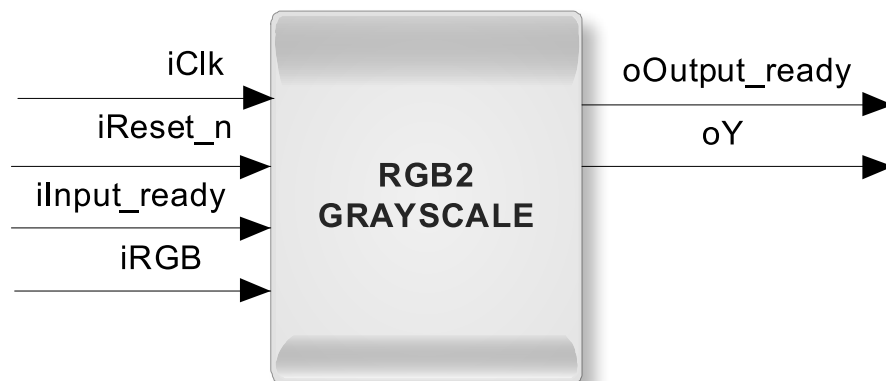
Hình 10: CONTROL_UNIT trong Khối 2D MFHWT

Tín hiệu	Độ rộng	Hướng	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iReset_n	1	Input	Tín hiệu Reset đồng bộ tác động mức thấp.
iData_ready	1	Input	Tín hiệu báo có dữ liệu đến khối 2D MFHWT.
iFull_Buffer0	8	Input	Tín hiệu báo FIFO đầy từ PPBUFFER_640x4.
iEmpty_Buffer0	2	Input	Tín hiệu báo FIFO rỗng từ PPBUFFER_640x4.
iRdready_Buffer1	1	Input	Tín hiệu báo dữ liệu sẵn sàng từ PPBUFFER.
iFull_Buffer2	4	Input	Tín hiệu báo FIFO đầy từ BUFFER_160x4.

iEmpty_Buffer2	1	Input	Tín hiệu báo FIFO rỗng từ BUFFER_160x4.
oOutput_ready	1	Output	Tín hiệu báo dữ liệu ra hợp lệ.
oSelect_Buffer0	1	Output	Tín hiệu chọn buffer tới PPBUFFER_640x4
oRdreq_Buffer0	2	Output	Tín hiệu yêu cầu đọc dữ liệu tới PPBUFFER_640x4
oWrreq_Buffer0	8	Output	Tín hiệu yêu cầu ghi dữ liệu tới PPBUFFER_640x4
oSelect_Buffer1	1	Output	Tín hiệu chọn register tới PPBUFFER_640x4
oWrreq_Buffer1	1	Output	Tín hiệu yêu cầu ghi dữ liệu tới PPBUFFER
oRdreq_Buffer2	1	Output	Tín hiệu yêu cầu đọc dữ liệu tới BUFFER_160x4
oWrreq_Buffer2	4	Output	Tín hiệu yêu cầu ghi dữ liệu tới BUFFER_160x4

2.1.2 Grayscale Convert:

Nhiệm vụ của khối Grayscale Convert là chuyển từ ảnh RGB 16bits sang ảnh xám (grayscale 8bits).



Hình 11: Grayscale Convert

Tín hiệu	Độ rộng	Ghi/ Đọc	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iReset_n	1	Input	Tín hiệu Reset đồng bộ tác động mức thấp.
iInput_ready	1	Input	Tín hiệu thông báo dữ liệu vào hợp lệ từ 2D MFHT.
iData_in	16	Input	Dữ liệu từ 2D MFHT đưa vào .
oOutput_ready	1	Output	Tín hiệu thông báo dữ liệu ra hợp lệ tới IIG.

oData_out	8	Output	Dữ liệu sau khi chuyển từ RGB => Y.
-----------	---	--------	-------------------------------------

2.1.2.1 Phép chuyển từ RGB sang YCbCr:

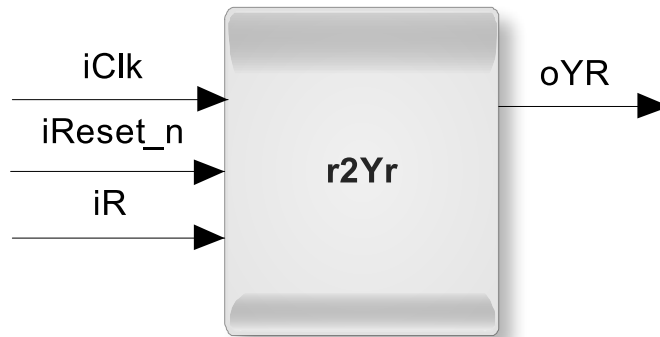
Để chuyển từ ảnh màu RGB sang YCbCr, ta dùng công thức chuyển đổi (1). Trong thiết kế, ta chỉ cần tính thành phần Y là được ảnh grayscale từ ảnh RGB.

$$\begin{cases} Y = 0.299 * R + 0.587 * G + 0.114 * B \\ Cb = -0.172 * R - 0.339 * G + 0.511B + 128 \\ Cr = 0.511R - 0.428 * G - 0.083 * B + 128 \end{cases} \quad (1)$$

2.1.2.2 Chi tiết thiết kế:

2.1.2.2.1 r2Yr:

r2Yr chuyển thành phần R sang YR ($YR = 0.299 * R$)

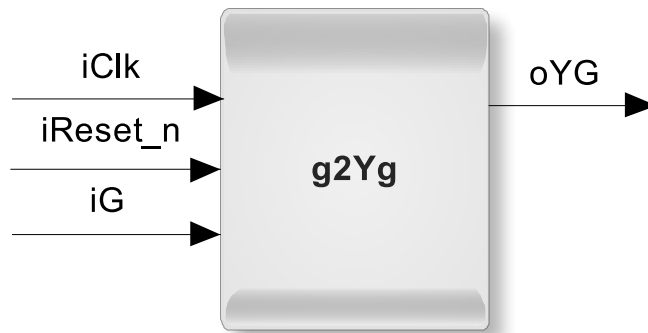


Hình 12: r2Yr trong Khối Grayscale Convert

Tín hiệu	Độ rộng	Ghi/ Đọc	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iReset_n	1	Input	Tín hiệu Reset đồng bộ tác động mức thấp.
iR	1	Input	Dữ liệu R của RGB
oYR	8	Output	Dữ liệu sau khi chuyển từ R => YR.

2.1.2.2.2 g2Yg:

g2Yg chuyển thành phần G sang YG ($YG = 0.587 * G$)

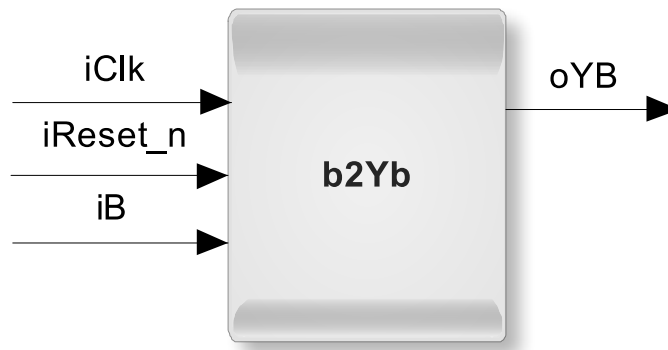


Hình 13: g2Yg trong Khối Grayscale Convert

Tín hiệu	Độ rộng	Ghi/ Đọc	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iReset_n	1	Input	Tín hiệu Reset đồng bộ tác động mức thấp.
iG	1	Input	Dữ liệu G của RGB
oYG	8	Output	Dữ liệu sau khi chuyển từ G => YG.

2.1.2.2.3 b2Yb:

b2Yb chuyển thành phần B sang YB ($YB = 0.114 \cdot B$)



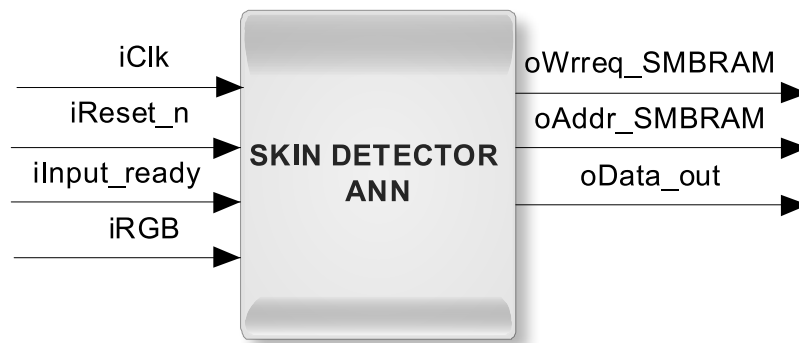
Hình 14: b2Yb trong Khối Grayscale Convert

Tín hiệu	Độ rộng	Ghi/ Đọc	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iReset_n	1	Input	Tín hiệu Reset đồng bộ tác động mức thấp.

iB	1	Input	Dữ liệu B của RGB
oYB	8	Output	Dữ liệu sau khi chuyển từ B => YB.

2.1.3 Skin Detector ANN:

Nhiệm vụ của khối Skin Detector ANN là kiểm tra pixel đầu vào có phải là màu da hay không. Nếu là màu da thì dữ liệu ngõ ra bằng 1 còn không phải thì dữ liệu ngõ ra bằng 0. Dữ liệu ngõ ra này sẽ được ghi vào Serach_Map_BRAM dùng như trigger để điều khiển bộ Detection_Core.



Hình 15: Skin Detector ANN trong Khối Pre_Processing

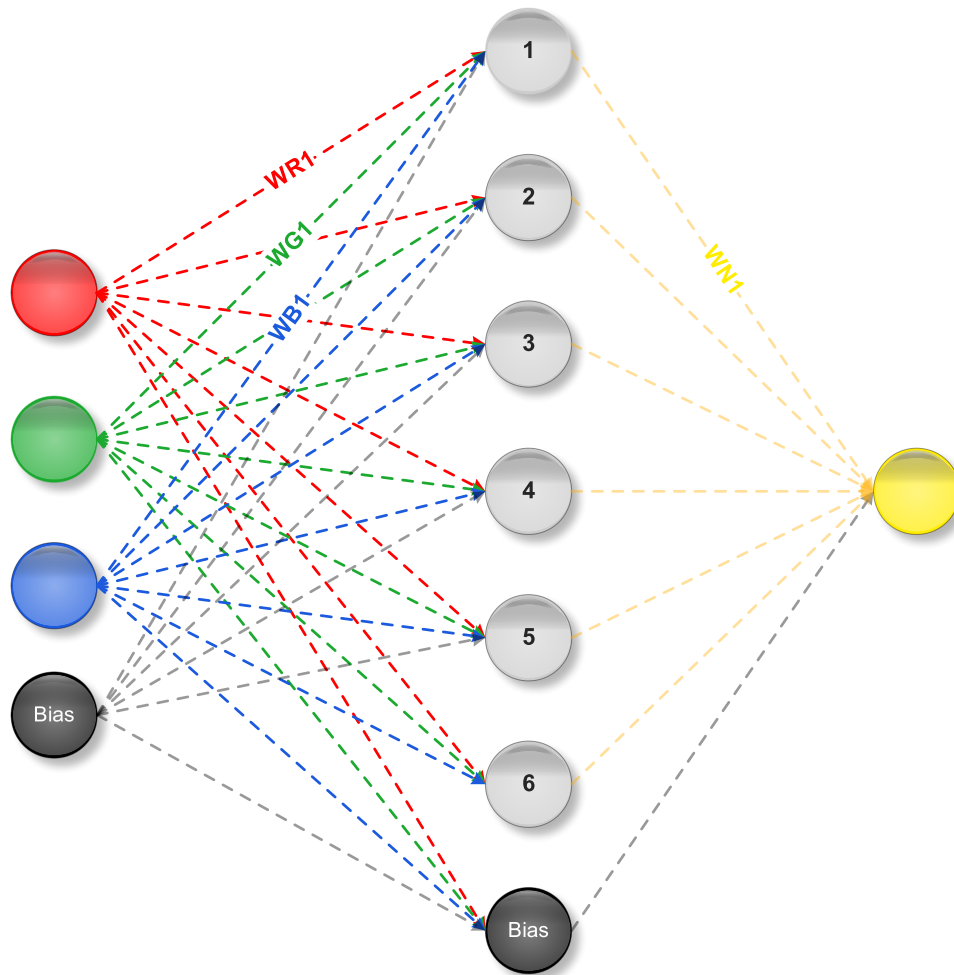
Tín hiệu	Độ rộng	Ghi/ Đọc	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iReset_n	1	Input	Tín hiệu Reset đồng bộ tác động mức thấp.
iInput_ready	1	Input	Tín hiệu báo dữ liệu vào hợp lệ từ 2D MFHWT.
iRGB	16	Input	Dữ liệu 1 pixel RGB từ 2D MFHWT vào.
oOutput_ready	1	Output	Tín hiệu báo dữ liệu ra hợp lệ.
oAddr_SMBRAM	13	Output	Địa chỉ Search_Map BRAM
oData_out	1	Output	Dữ liệu sau khi skin detect {0,1}.

2.1.3.1 Thuật toán Skin Detection ANN:

Trong thiết kế này, ta sẽ sử dụng thuật toán Artificial Neural Network để kiểm tra pixel có phải màu da hay không. ANN là một mô hình toán học, được mô phỏng như một mạng thần kinh sinh học, có khả năng thích ứng với các dữ liệu mẫu đầu vào sao cho dữ liệu đầu ra phù hợp với yêu cầu thông qua quá trình học. Nó thường được áp dụng trong các bài toán nhận dạng mẫu. Để rõ hơn về quá trình hoạt động của ANN, chúng ta sẽ xét

1 ví dụ cụ thể, đó là bài toán xác định màu da. Mô hình Skin Detection ANN trong thiết kế sẽ có 3 layer:

- Input layer : R, G, B, bias(1)
- Hidden layer: 6 neuron, bias(1)
- Output layer : 1neuron



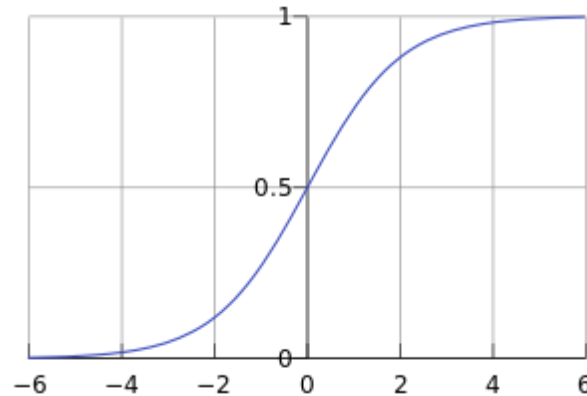
Hình 16: Skin Detector ANN

Công thức tại mỗi Hidden neuron:

$$x_i = R * W_{Ri} + G * W_{Gi} + B * W_{Bi} + 1 * W_{bias} \quad (2)$$

với $i = 1, \dots, 6$; $W_{\{R,G,B\}i}$: là trọng số của từng neuron; $bias = 1$

$$N_i = f(x_i) = \frac{1}{1 + e^{-x_i}} \quad (3) : \text{Sigmoid function}$$



Hình 17: Đồ thị hàm Sigmoid Fuction

Tại Output neuron áp dụng (4) và (3):

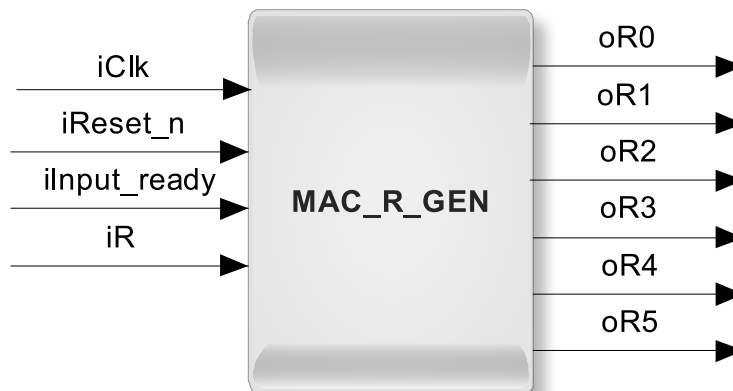
$$X = N_1 * W_{N1} + N_2 * W_{Ni} + N_3 * W_{N3} + N_4 * W_{N4} + \\ N_5 * W_{N5} + N_6 * W_{N6} + 1 * W_{bias} \quad (4)$$

Ở đây, các W: là trọng số có được sau khi cho ANN học các tập mẫu. Thuật toán học Backpropagation được áp dụng trong việc tính toán các trọng số này (có trong Matlab).

2.1.3.2 Chi tiết thiết kế;

2.1.3.2.1 MAC_R_GEN:

Nhiệm vụ của MAC_R_GEN là nhân thành phần R với 6 trọng số của 6 hidden neuron.

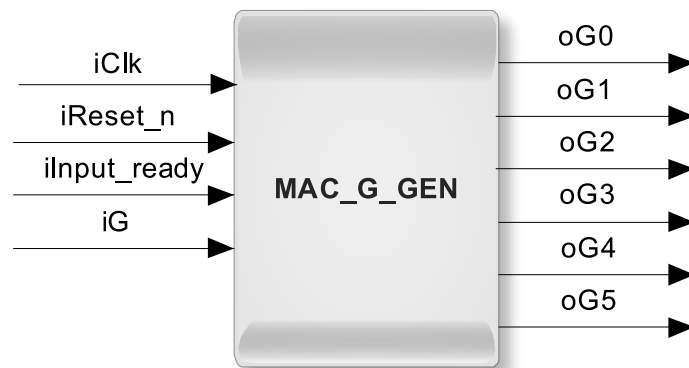


Hình 18: MAC_R_GEN trong Khối Skin Detector ANN

Tín hiệu	Độ rộng	Ghi/ Đọc	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iReset_n	1	Input	Tín hiệu Reset đồng bộ tác động mức thấp.
iInput_ready	1	Input	Tín hiệu thông báo dữ liệu vào hợp lệ.
iR	8	Input	Dữ liệu R đưa vào .
oR0....oR5	29	Output	Dữ liệu ra sau khi nhân R lần lượt với 6 trọng số (W).

2.1.3.2.2 MAC_G_GEN:

Nhiệm vụ của MAC_G_GEN là nhân thành phần G với 6 trọng số của 6 hidden neuron.

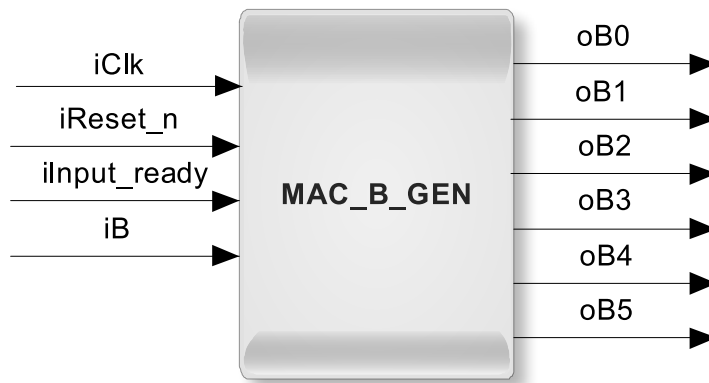


Hình 19: MAC_G_GEN trong Khối Skin Detector ANN

Tín hiệu	Độ rộng	Ghi/ Đọc	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iReset_n	1	Input	Tín hiệu Reset đồng bộ tác động mức thấp.
iInput_ready	1	Input	Tín hiệu thông báo dữ liệu vào hợp lệ.
iR	8	Input	Dữ liệu R đưa vào .
oR0....oR5	29	Output	Dữ liệu ra sau khi nhân R lần lượt với 6 trọng số (W).

2.1.3.2.3 MAC_B_GEN:

Nhiệm vụ của MAC_B_GEN là nhân thành phần B với 6 trọng số của 6 hidden neuron.

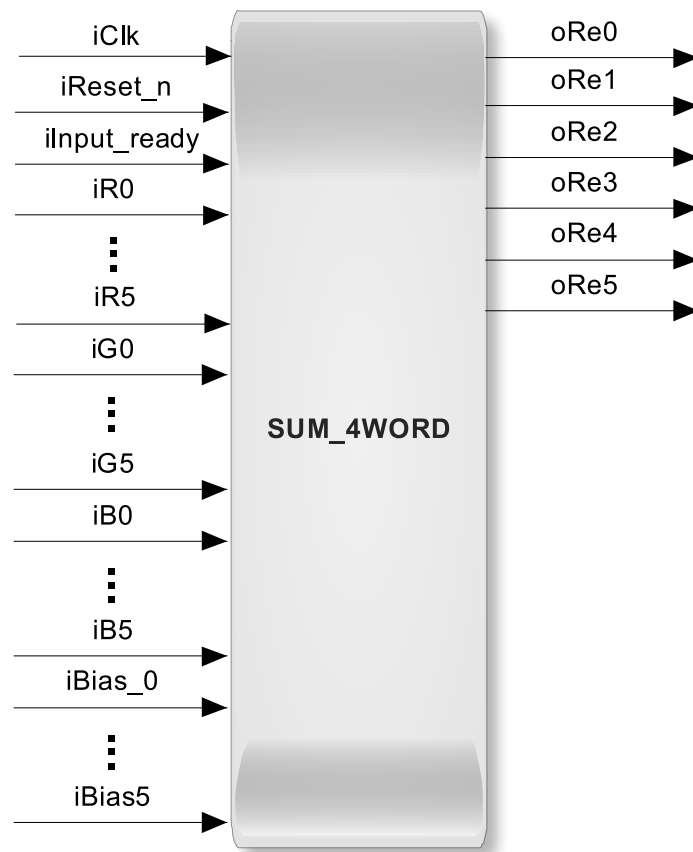


Hình 20: MAC_B_GEN trong Khối Skin Detector ANN

Tín hiệu	Độ rộng	Ghi/ Đọc	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iReset_n	1	Input	Tín hiệu Reset đồng bộ tác động mức thấp.
iInput_ready	1	Input	Tín hiệu thông báo dữ liệu vào hợp lệ.
iR	8	Input	Dữ liệu R đưa vào .
oR0....oR5	29	Output	Dữ liệu ra sau khi nhân R lần lượt với 6 trọng số (W).

2.1.3.2.4 SUM_4WORD:

SUM_4WORD cộng từng bộ $(R*W + G*W + B*W + 1*W)$ với $R*W$, $G*W$, $B*W$ là dữ liệu ra của MAC_R_GEN, MAC_G_GEN, MAC_B_GEN.



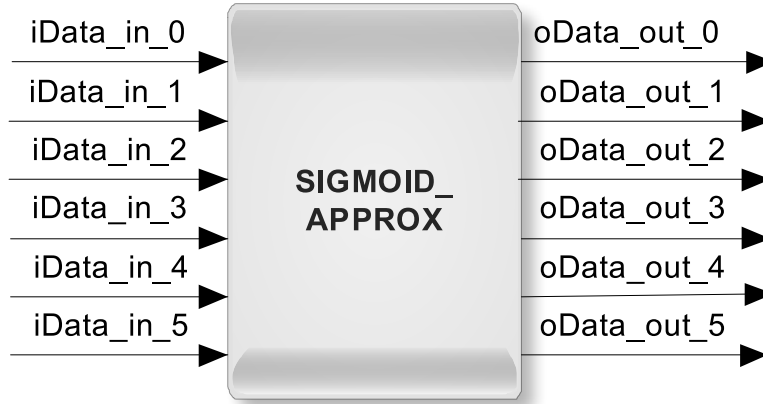
Hình 21: SUM_4SWORD trong Khối Skin Detector ANN

Tín hiệu	Độ rộng	Ghi/ Đọc	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iReset_n	1	Input	Tín hiệu Reset đồng bộ tác động mức thấp.
iInput_ready	1	Input	Tín hiệu thông báo dữ liệu vào hợp lệ.
iR0...iR5	29x6	Input	Dữ liệu $R*W$ từ MAC_R_GEN đưa vào.
iG0...iG5	29x6	Input	Dữ liệu $G*W$ từ MAC_G_GEN đưa vào.
iB0...iB5	29x6	Input	Dữ liệu $B*W$ từ MAC_B_GEN đưa vào.
iBias_0...iBias_5	29x6	Input	Dữ liệu bias được gán trong các thanh ghi.
oRe0....oRe5	29x6	Output	Dữ liệu ra sau áp dụng (2).

2.1.3.2.5 SIGMOID_APPROX:

SIGMOID_APPROX tính xấp xỉ công thức (3) Sigmoid function cho các hidden neuron theo công thức sau:

$$f(x) = \begin{cases} 1 & \text{nếu } x \geq 0 \\ 0 & \text{nếu } x < 0 \end{cases}$$

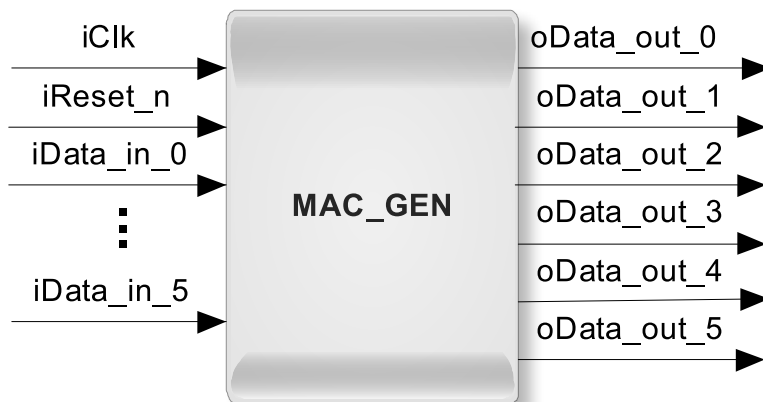


Hình 22: SIGMOID_APPROX trong Khối Skin Detector ANN

Tín hiệu	Độ rộng	Ghi/ Đọc	Mô tả
iData_in_0...iData_in_5	29x6	Input	Dữ liệu từ SUM_4WORD đưa vào.
oData_out_0...oData_out_5	1x6	Output	Dữ liệu ra sau khi tính xấp xỉ (output của hidden neuron).

2.1.3.2.6 MAC_GEN:

Nhiệm vụ của MAC_GEN là nhân 6 dữ liệu ngõ ra của SIGMOID_APPROX với 6 trọng số ở output neuron.

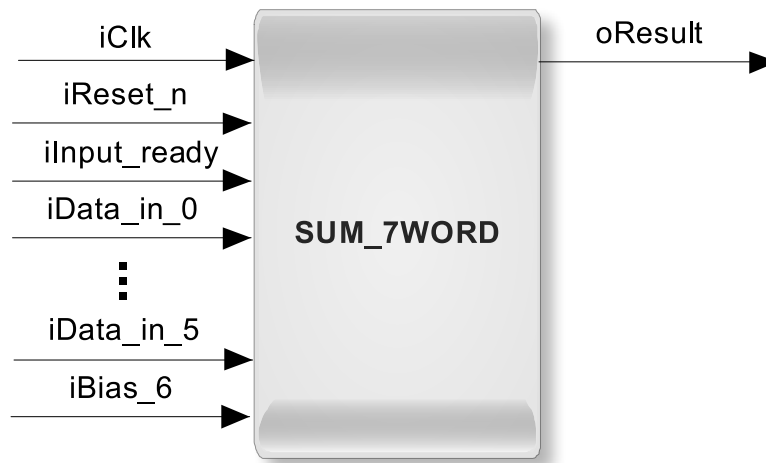


Hình 23: MAC_GEN trong Khối SKin Detector ANN

Tín hiệu	Độ rộng	Ghi/ Đọc	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iReset_n	1	Input	Tín hiệu Reset đồng bộ tác động mức thấp.
iData_in_0...iData_in_5	1x6	Input	Dữ liệu từ SIGMOID_APPROX đưa vào.
oData_out_0...oData_out_5	22x6	Output	Dữ liệu ra sau khi nhân các output hidden neuron với các trọng số tương ứng.

2.1.3.2.7 SUM_7WORD:

SUM_7WORD cộng 6 dữ liệu ngõ ra của MAC_GEN và output bias như công thức (4).



Hình 24: SUM_7WORD

Tín hiệu	Độ rộng	Ghi/ Đọc	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iReset_n	1	Input	Tín hiệu Reset đồng bộ tác động mức thấp.
iData_in_0...iData_in_5	21x6	Input	Dữ liệu từ MAC_GEN đưa vào.
iBias_6	21	Input	Dữ liệu bias được gắn trong thanh ghi.
oResult	23	Output	Dữ liệu ra sau khi áp dụng (3).

2.1.3.2.8 SIGMOID_APPROX_1:

SIGMOID_APPROX_1 tính xấp xỉ công thức (3) Sigmoid function cho output neuron theo công thức sau:

$$f(x) = \begin{cases} 1 & \text{nếu } x \geq 3.1886 \\ 0 & \text{nếu } x \leq 3.2018 \end{cases}$$

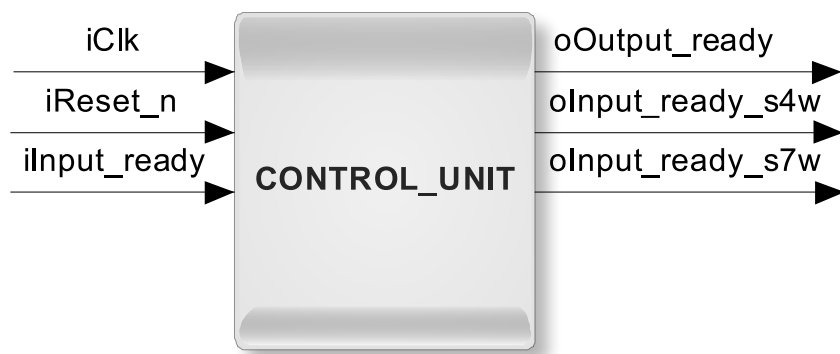


Hình 25: SIGMOID_APPROX_1 trong Khối Skin Detector ANN

Tín hiệu	Độ rộng	Ghi/ Đọc	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iReset_n	1	Input	Tín hiệu Reset đồng bộ tác động mức thấp.
iData_in_0	21	Input	Dữ liệu từ SIGMOID_APPROX_1 đưa vào.
oData_out_0	1	Output	Dữ liệu ra sau khi tính xấp xỉ (output của output neuron).

2.1.3.2.9 CONTROL_UNIT:

CONTROL_UNIT điều khiển khối Skin Detector ANN hoạt động đồng bộ. Chủ yếu là truyền các tín hiệu báo dữ liệu hợp lệ.

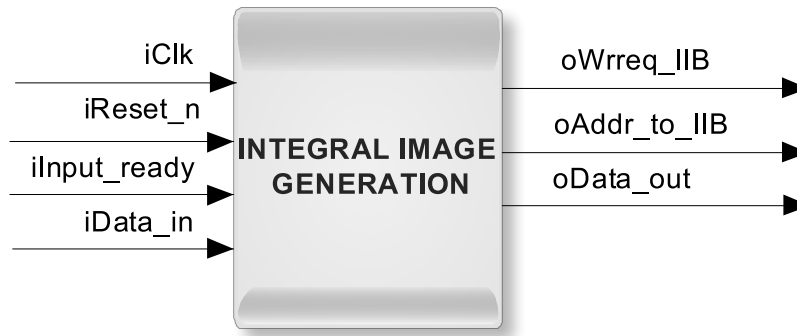


Hình 26: CONTROL_UNIT trong Khối Skin Detector ANN

Tín hiệu	Độ rộng	Ghi/ Đọc	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iReset_n	1	Input	Tín hiệu Reset đồng bộ tác động mức thấp.
iInput_ready	1	Input	Tín hiệu thông báo dữ liệu vào Skin Detector ANN hợp lệ.
oOutput_ready	1	Output	Tín hiệu thông báo dữ liệu ra hợp lệ.
oInput_ready_s4w	1	Output	Tín hiệu thông báo dữ liệu vào SUM_4SWORD hợp lệ.
oInput_ready_s7w	1	Output	Tín hiệu thông báo dữ liệu vào SUM_7WORD hợp lệ.

2.1.4 Integral Image Generation:

Khối IIG dùng để tính toán ảnh tích hợp (integral image), ảnh tích hợp này sẽ giúp tính nhanh các Haar feature.

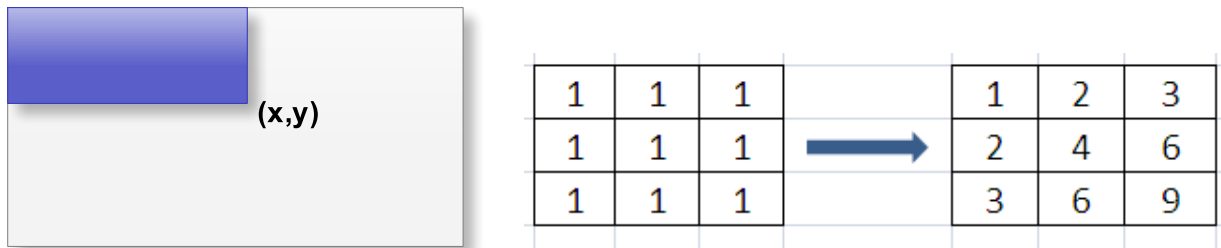


Hình 27: IIG trong Khối Pre_Processing

Tín hiệu	Độ rộng	Ghi/ Đọc	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iReset_n	1	Input	Tín hiệu Reset đồng bộ tác động mức thấp.
iInput_ready	1	Input	Tín hiệu thông báo dữ liệu vào hợp lệ từ Grayscale Convert.
iData_in	8	Input	Dữ liệu từ Grayscale Convert đưa vào .
oWrreq_IIB	1	Output	Tín hiệu yêu cầu ghi dữ liệu tới Integral Image BRAM.
oAddr_to_IIB	13	Output	Địa chỉ vùng nhớ Integral Image BRAM để ghi dữ liệu vào.
oData_out	21	Output	Dữ liệu sau khi tính toán.

2.1.4.1 Khái niệm và phương pháp tính Integral Image:

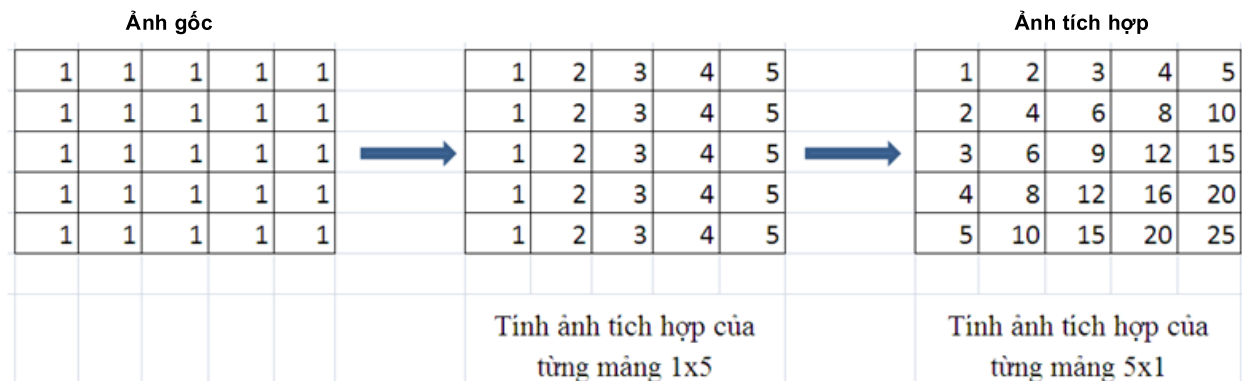
Ảnh tích hợp là 1 khái niệm chỉ tổng giá trị pixel của hình ảnh gốc, giá trị tại 1 điểm (x,y) bất kì bằng tổng các pixel của phần hình chữ nhật phía trên bên trái. Cấu trúc hình ảnh tích hợp như hình 28 .



Hình 28: Cấu trúc Integral Image

Tổng các pixel ảnh gốc trong phần hình chữ nhật màu xanh là giá trị tại điểm (x,y) trên ảnh tích hợp. 2 mảng bên phải là ví dụ cụ thể 1 ảnh 3x3 và ảnh tích hợp của nó.

Phương pháp tính nhanh ảnh tích hợp sẽ được mô tả chi tiết như sau (ví dụ ta có ảnh 5x5):

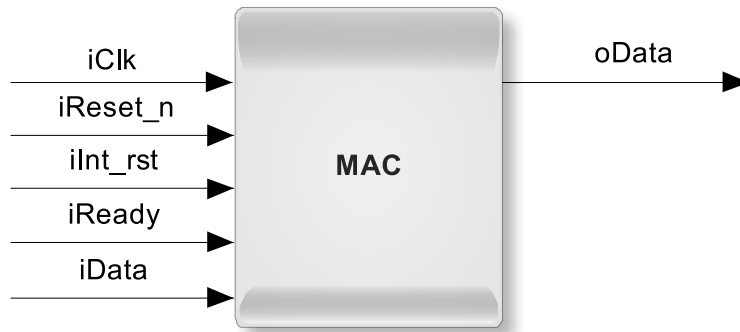


Hình 29: Phương pháp tính Integral Image

2.1.4.2 Chi tiết thiết kế:

2.1.4.2.1 MAC:

MAC gồm 1 bộ cộng và 1 register. Khi có dữ liệu vào bộ cộng sẽ cộng giá trị mới vào với giá trị trước đó lưu trong thanh ghi. Với ảnh $M \times N$, ta xem như là 1 mảng $M \times N$ phần tử và được đưa vào IIG theo trình tự từ trái sang phải từ trên xuống dưới.

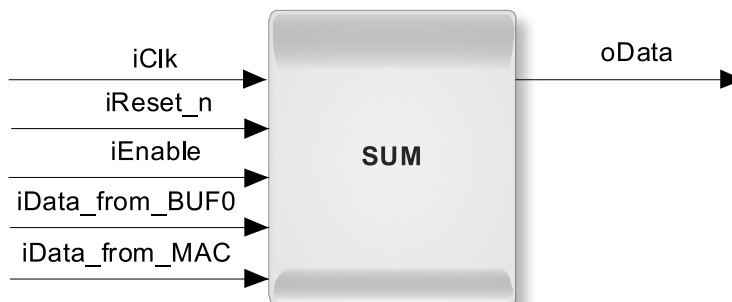


Hình 30: MAC trong Khối IIG

Tín hiệu	Độ rộng	Ghi/ Đọc	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iReset_n	1	Input	Tín hiệu Reset đồng bộ tác động mức thấp.
iInt_rst	1	Input	Tín hiệu reset thanh ghi từ CONTROL_UNIT khi đã xong 1 hàng hay 1 cột.
iReady	1	Input	Tín hiệu thông báo dữ liệu vào hợp lệ.
iData	8	Input	Dữ liệu 1 pixel grayscale.
oData	21	Output	Dữ liệu sau khi cộng dồn với các giá trị trước.

2.1.4.2.2 SUM:

Cộng giá trị ra từ MAC với giá trị ,có cùng cột và hàng tăng 1 đơn vị so với giá trị vào MAC, được lưu trong PPBUFFER_80x1.



Hình 31: SUM trong Khối IIG

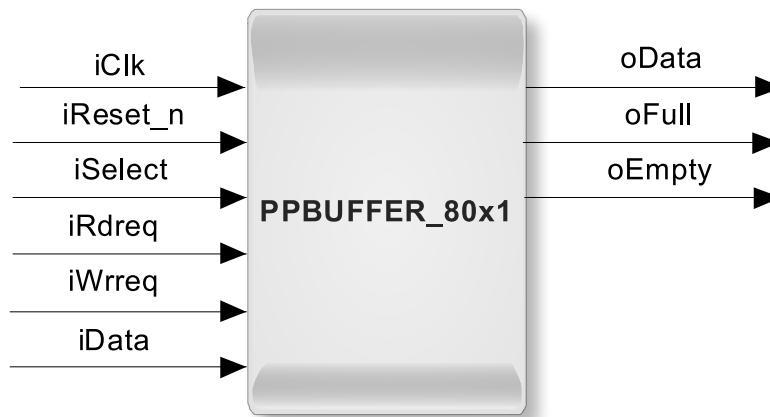
Tín hiệu	Độ rộng	Ghi/ Đọc	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iReset_n	1	Input	Tín hiệu Reset đồng bộ tác động mức thấp.
iEnable	1	Input	Tín hiệu cho phép cộng từ CONTROL_UNIT.
iData_from_BUF0	21	Input	Dữ liệu từ PPBUFFER_80x1 vào.
iData_from_MAC	21	Input	Dữ liệu từ MAC vào.
oData	21	Output	Dữ liệu sau khi cộng.

2.1.4.2.3 SCFIFO_80x1:

Thiết kế như phần 2.1.1.2.2.1:

2.1.4.2.4 PPBUFFER_80x1:

Đây là 1 pingpong buffer dùng để lưu dữ liệu của 1 dòng sau khi đã tính theo công thức ảnh tích hợp trên 1 dòng ảnh (như ví dụ hình). Khi có dữ liệu của dòng sau, buffer(dùng FIFO) sẽ đẩy dữ liệu ra cộng với giá trị MAC tính được sẽ được giá trị tại 1 điểm trên ảnh tích hợp của ảnh gốc (như ví dụ hình).



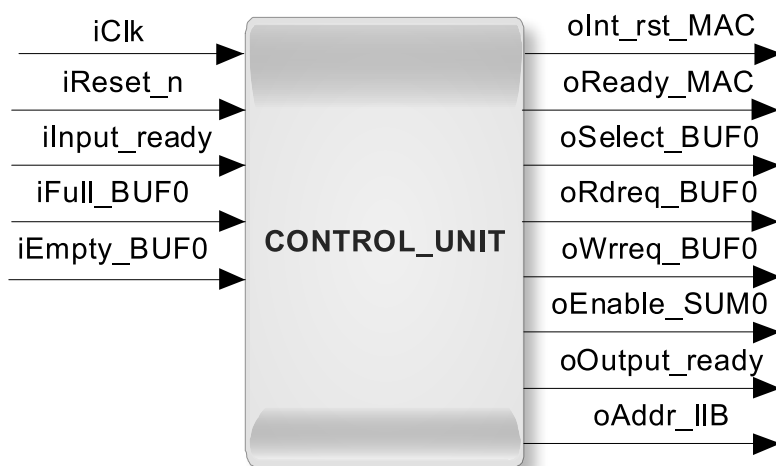
Hình 32: PPBUFFER_80x1 trong Khối IIG

Tín hiệu	Độ rộng	Hướng	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iSelect	1	Input	Tín hiệu chọn buffer bên trong để đọc và ghi.
iData	21	Input	Dữ liệu đưa vào PPBUFFER_80x1 từ SUM.
iRdreq	1	Input	Tín hiệu yêu cầu đọc dữ liệu từ PPBUFFER_80x1.
iWrreq	1	Input	Tín hiệu yêu cầu ghi dữ liệu tới PPBUFFER_80x1.

oFull	1	Output	Tín hiệu báo trạng thái đầy của FIFO bên trong.
oEmpty	1	Output	Tín hiệu báo trạng thái rỗng của FIFO bên trong.
oData	21	Output	Dữ liệu lấy ra từ PPBUFFER_80x1 tới SUM.

2.1.4.2.5 CONTROL_UNIT:

Điều khiển IIG hoạt động đồng bộ.

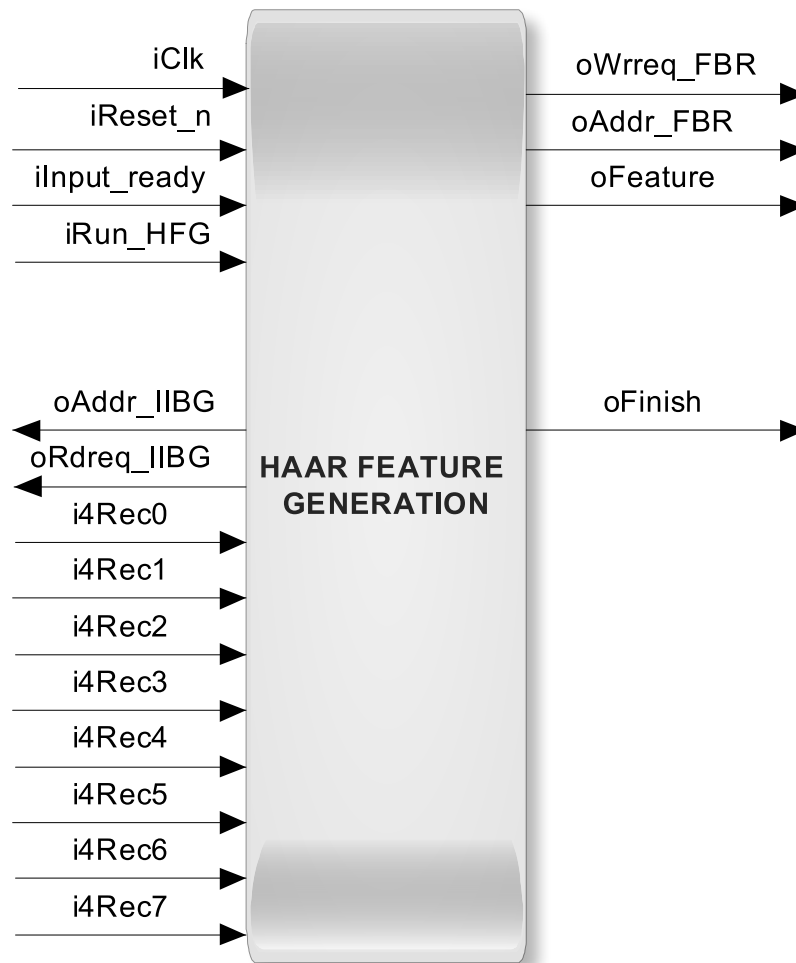


Hình 33: CONTROL_UNIT trong Khối IIG

Tín hiệu	Độ rộng	Hướng	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iReset_n	1	Input	Tín hiệu Reset đồng bộ tác động mức thấp.
iInput_ready	1	Input	Tín hiệu thông báo dữ liệu vào hợp lệ đến IIG
iFull_BUF0	1	Input	Tín hiệu báo đầy từ PPBUFFER_80x1.
iEmpty_BUF0	1	Input	Tín hiệu báo rỗng từ PPBUFFER_80x1.
oInt_rst_MAC	1	Output	Tín hiệu reset thanh ghi của MAC.
oReady_MAC	1	Output	Tín hiệu báo dữ liệu hợp lệ tới MAC.
oSelect_BUF0	1	Output	Tín hiệu chọn buffer bên trong PPBUFER_80x1 để đọc và ghi.
oRdreq_BUF0	1	Output	Tín hiệu yêu cầu đọc dữ liệu từ PPBUFFER_80x1.
oWrreq_BUF0	1	Output	Tín hiệu yêu cầu ghi dữ liệu tới PPBUFFER_80x1.
oEnable_SUM0	1	Output	Tín hiệu cho phép SUM thực thi.
oOutput_ready	1	Output	Tín hiệu báo dữ liệu ra hợp lệ.

2.2 Detection_Core:

2.2.1 Haar Feature Generation:



Hình 34: HAAR FEATURE GENERATION trong Khối Detection

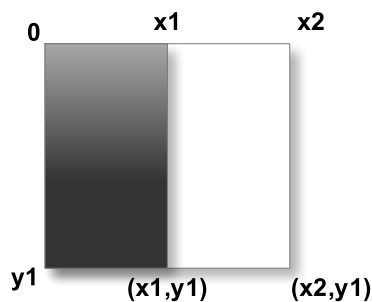
Tín hiệu	Độ rộng	Ghi/ Đọc	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iReset_n	1	Input	Tín hiệu Reset đồng bộ tác động mức thấp.
iInput_ready	1	Input	Tín hiệu thông báo dữ liệu vào hợp lệ từ Grayscale Convert.
iRun_HFG	1	Input	Tín hiệu cho phép Detection Core thực thi.
i4Rec0	84	Input	Dữ liệu từ Integral Image BRAM Group

			đưa vào.
i4Rec1	84	Input	Dữ liệu từ Integral Image BRAM Group đưa vào.
i4Rec2	84	Input	Dữ liệu từ Integral Image BRAM Group đưa vào.
i4Rec3	84	Input	Dữ liệu từ Integral Image BRAM Group đưa vào.
i4Rec4	84	Input	Dữ liệu từ Integral Image BRAM Group đưa vào.
i4Rec5	84	Input	Dữ liệu từ Integral Image BRAM Group đưa vào.
i4Rec6	84	Input	Dữ liệu từ Integral Image BRAM Group đưa vào.
i4Rec7	84	Input	Dữ liệu từ Integral Image BRAM Group đưa vào.
oRdreq_IIBG	36	Output	Tín hiệu yêu cầu đọc dữ liệu từ IIBG.
oAddr_IIBG	1	Output	Địa chỉ vùng nhớ IIBG cần đọc dữ liệu.
oWrreq_FBR	1	Output	Tín hiệu yêu cầu ghi dữ liệu từ HFG tới Feature BRAM.
oAddr_FBR	7	Output	Địa chỉ vùng nhớ Feature BRAM để ghi dữ liệu vào.
oFeature	22	Output	Dữ liệu của 1 haar feature.
oFinish	1	Output	Tín hiệu báo cho Detection_Controller biết là đã tính xong 1 feature.

2.2.1.1 Khái niệm và phương pháp tính các đặc trưng Haar:

Đặc trưng Haar là 1 hình bao gồm các hình chữ nhật đen, trắng dùng để xác định khuôn mặt trong ảnh. Mỗi đặc trưng là 1 bộ lọc yếu, chúng sẽ được kết hợp thành bộ lọc mạnh thông qua thuật toán train Adaboost. Mỗi đặc trưng được tính bằng cách tìm độ chênh lệch giữa tổng các pixel của hình chữ trắng và đen. Để tính tổng pixel của các hình chữ nhật, ta sử dụng Integral Image để tính nhanh chúng. Sau đây là ví dụ cụ thể:

Giả sử, ta có 1 đặc trưng Haar như sau:



$$\begin{cases} S_B = I_{(x1,y1)} \\ S_W = I_{(x2,y1)} - I_{(x1-1,y1-1)} \end{cases} \rightarrow F = S_W - S_B$$

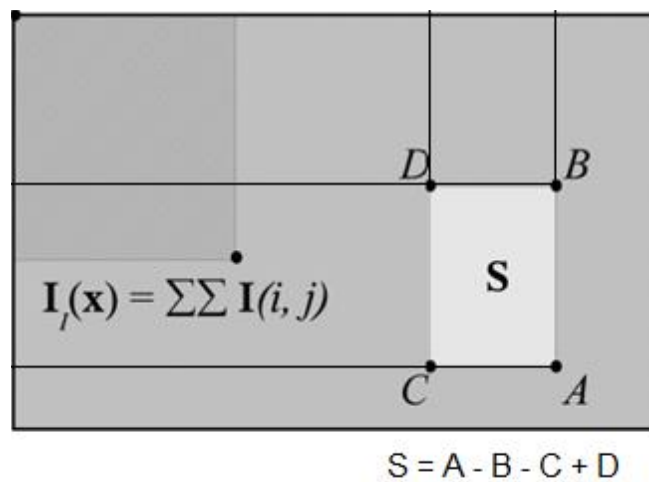
với S_B : tổng pixel của hcn đen;

S_W : tổng pixel của hcn trắng;

F : giá trị của đặc trưng Haar

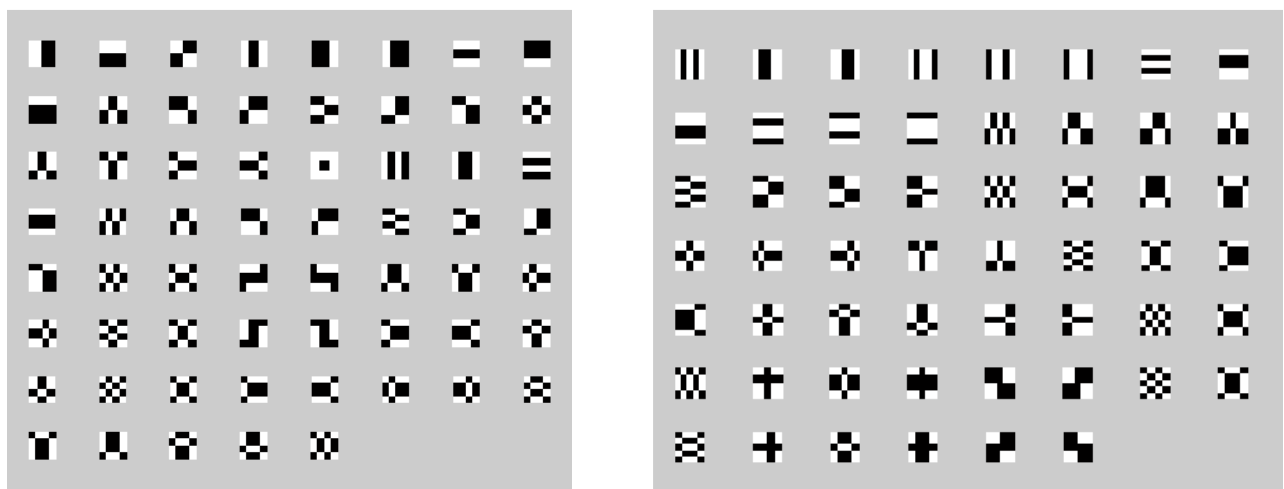
$I(x,y)$: giá trị tại điểm (x,y) trên ảnh tích hợp

Công thức tổng quát để tính 1 hcn dựa vào Integral Image:



Hình 35: Công thức tính hình chữ nhật dựa vào integral image

Danh sách 115 đặc trưng Haar được dùng trong thiết kế:

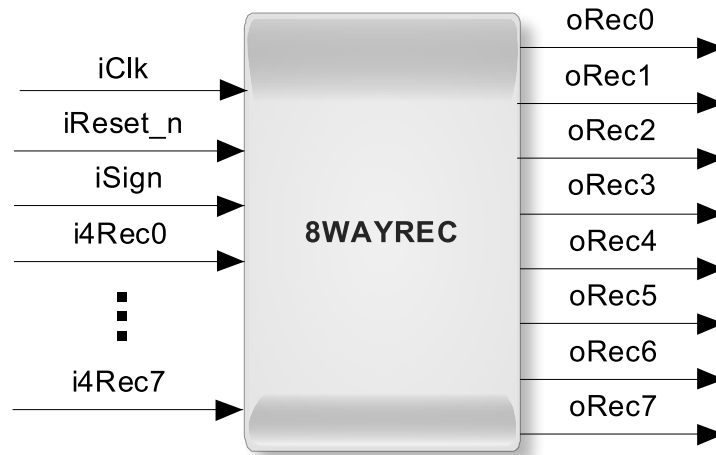


Hình 36: 115 đặc trưng Haar

2.2.1.2 Chi tiết thiết kế:

2.2.1.2.1 8WAYREC:

8WAYREC tính song song 8 hình chữ nhật của 1 feature cùng lúc.

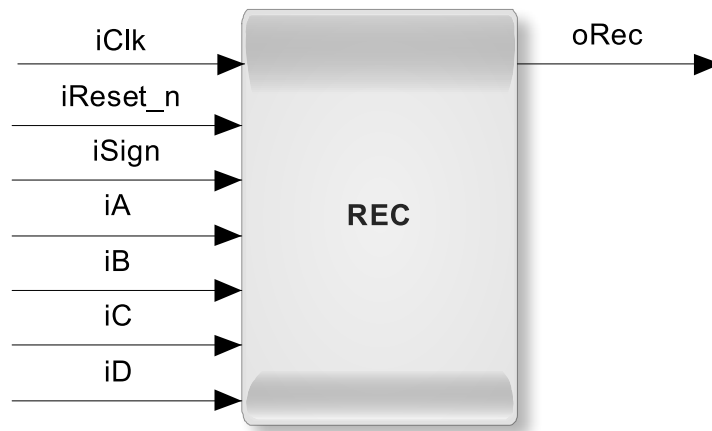


Hình 37: 8WAYREC trong Khối HFG

Tín hiệu	Độ rộng	Ghi/ Đọc	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iReset_n	1	Input	Tín hiệu Reset đồng bộ tác động mức thấp.
iSign	8	Input	Dữ liệu bit dấu của các hcn (1: đen, 0: trắng)
i4Rec0...i4Rec7	84x8	Input	Dữ liệu từ Integral Image BRAM Group đưa vào.
oRec0...oRec7	21x8	Output	Dữ liệu ra sau khi tính các hcn dựa vào integral image.

2.2.1.2.1.1 REC:

REC là 1 bộ tính hình chữ nhật của 8WAYREC.

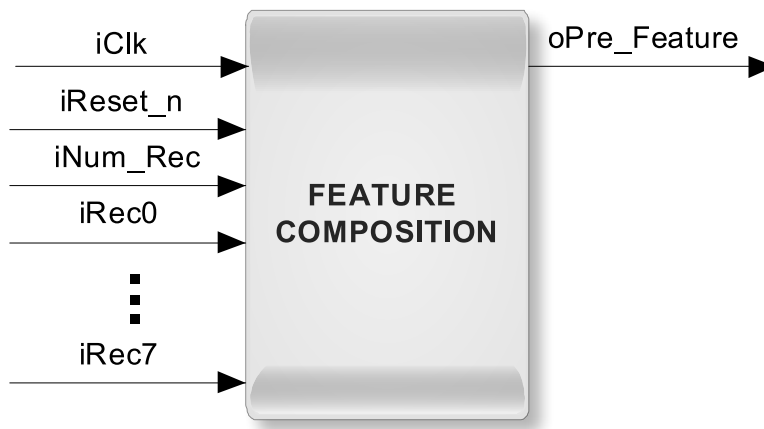


Hình 38: REC trong Khối 8WAYREC

Tín hiệu	Độ rộng	Ghi/ Đọc	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iReset_n	1	Input	Tín hiệu Reset đồng bộ tác động mức thấp.
iSign	8	Input	Dữ liệu bit dấu của hcn đang tính (1: đen, 0: trắng)
iA	21	Input	Dữ liệu vào như mô tả ở hình 35.
iB	21	Input	Dữ liệu vào như mô tả ở hình 35.
iC	21	Input	Dữ liệu vào như mô tả ở hình 35.
iD	21	Input	Dữ liệu vào như mô tả ở hình 35.
oRec	21	Output	Dữ liệu ra sau khi tính dựa vào integral image và xét bit dấu.

2.2.1.2.2 FEATURE COMPOSITION:

Các hcn sau khi tính xong sẽ qua FEATURE COMPOSITION để kết hợp thành 1 feature. Nhiệm vụ của FC là tính sự chênh lệch giá trị các pixel giữa các hcn trắng và hcn đen.



Hình 39: FEATURE COMPOSITION trong Khối HFG

Tín hiệu	Độ rộng	Ghi/ Đọc	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iReset_n	1	Input	Tín hiệu Reset đồng bộ tác động mức thấp.
iNum_Rec	4	Input	Dữ liệu về số lượng hcn trong feature đang tính.
iRec0...iRec7	21x8	Input	Dữ liệu từ 8WAYREC vào.
oPre_Feature	19	Output	Dữ liệu ra sau khi tính chênh lệch giữa các hcn trắng và các hcn đen.

2.2.1.2.3 NORMALIZATION:

Feature sau khi tính xong phải được chuẩn hóa bằng công thức sau:

$$F = \frac{Pre_F}{255 * W_x * W_y} \text{ với } W_x * W_y: \text{diện tích cửa sổ detect}$$

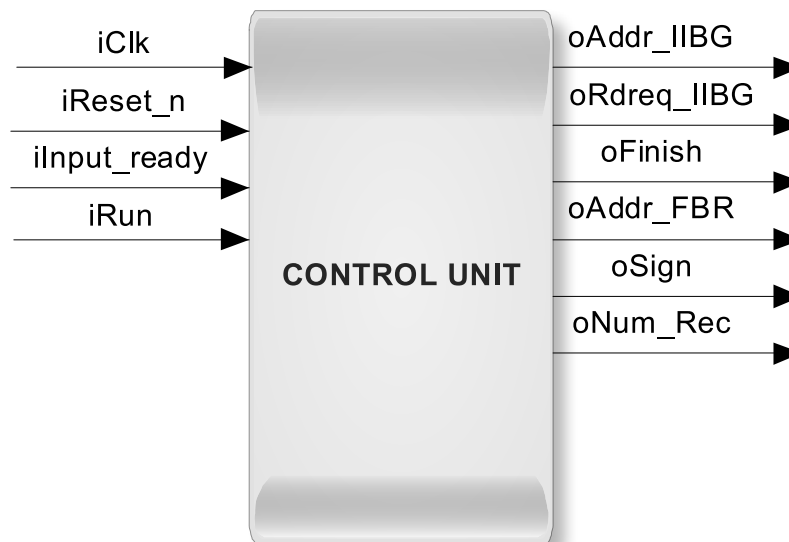


Hình 40: NORMALIZATION trong Khối HFG

Tín hiệu	Độ rộng	Ghi/ Đọc	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iReset_n	1	Input	Tín hiệu Reset đồng bộ tác động mức thấp.
iPre_feature	19	Input	Dữ liệu từ FEATURE_POSITION vào.
oFeature	22	Output	Dữ liệu ra sau khi chuẩn hóa feature.

2.2.1.2.4 CONTROL UNIT:

Nhận tín hiệu điều khiển từ DETECTION_CONTROL_UNIT và điều khiển các khối con trong HFG hoạt động đồng bộ.



Hình 41: CONTROL_UNIT trong Khối HFG

Tín hiệu	Độ rộng	Ghi/ Đọc	Mô tả
iClk	1	Input	Tín hiệu xung nhịp giúp toàn bộ khối hoạt động đồng bộ với nhau.
iReset_n	1	Input	Tín hiệu Reset đồng bộ tác động mức thấp.
iInput_ready	1	Input	Tín hiệu thông báo dữ liệu vào hợp lệ từ Grayscale Convert.
iRun	1	Input	Tín hiệu cho phép Detection Core thực thi.
oRdreq_IIBG	36	Output	Tín hiệu yêu cầu đọc dữ liệu từ IIBG.
oAddr_IIBG	1	Output	Địa chỉ vùng nhớ IIBG cần đọc dữ liệu.
oAddr_FBR	7	Output	Địa chỉ vùng nhớ Feature BRAM để ghi dữ liệu vào.
oFeature	22	Output	Dữ liệu của 1 haar feature.
oFinish	1	Output	Tín hiệu báo cho Detection_Controller biết là đã tính xong 1 feature.
iSign	8	Output	Dữ liệu bit dấu của 8 hcn đầu vào (1: đen, 0: trắng)
iNum_Rec	4	Output	Dữ liệu về số lượng hcn trong feature đang tính.

