



THIẾT KẾ MỘT BỘ ĐIỀU KHIỂN TRÒ CHƠI DỰA TRÊN SỰ DI CHUYỂN CỦA BÀN TAY TRÊN NỀN FPGA

Trương Phong Tuyên¹, Trần Thanh Sang², Trần Trung Hiếu² và Lương Vinh Quốc Danh¹

¹ Khoa Công nghệ, Trường Đại học Cần Thơ

² Ngành Kỹ thuật Máy tính, Khóa 36, Khoa Công nghệ, Trường Đại học Cần Thơ

Thông tin chung:

Ngày nhận: 19/12/2014

Ngày chấp nhận: 24/04/2015

Title:

Design and implementation of an FPGA-based hand-tracking game controller

Từ khóa:

Altera DE2-115, FPGA, hệ thống nhúng, theo dõi bàn tay, xử lý ảnh

Keywords:

Altera DE2-115, embedded systems, FPGA, hand-tracking, image processing

ABSTRACT

In recent years, embedded system is an outstanding option to carry out almost all of natural interactive control applications. Besides, it is a contemporary trend towards utilizing natural user interfaces such as head or hand gesture interaction, etc. Existing software solutions; however, still show weak points in processing speed, especially in real-time applications. Therefore, this research proposed a hardware solution by implementing an embedded system on FPGA. This embedded system was built on Altera's DE2-115 board with input from a 5-Mega pixel camera, which was able to recognize and track hand movement to handle a computer game simultaneously. Preliminary results encourage further research on FPGA-based embedded systems for smart interaction applications.

TÓM TẮT

Trong những năm gần đây, các ứng dụng sử dụng phương pháp điều khiển giao diện tự nhiên như thông qua cử động của đầu, tay... đang là một xu thế. Tuy nhiên, các giải pháp phần mềm cho việc điều khiển trên hiện tại vẫn còn hạn chế về tốc độ xử lý, đặc biệt là trong các ứng dụng thời gian thực. Từ đó, trên cơ sở thực hiện hệ thống nhúng trên nền FPGA, nghiên cứu này đề xuất một giải pháp phần cứng cho vấn đề đã nêu. Hệ thống được thực hiện sử dụng board Altera DE2-115 và một camera có độ phân giải 5 Mega pixel cho phép phát hiện và bám theo sự di chuyển của bàn tay người để điều khiển một trò chơi một cách tức thời. Kết quả của nghiên cứu là cơ sở cho việc tiếp tục nghiên cứu, phát triển các ứng dụng nhúng tương tác thông minh trên nền FPGA.

1 GIỚI THIỆU

Sự cạnh tranh trong kinh doanh ngày càng tăng, đặc biệt đối với ngành công nghiệp giải trí do đó nhu cầu về các phương pháp điều khiển thân thiện, thông minh góp phần quan trọng đến sự thành công của một sản phẩm mới. Chính vì thế, kỹ thuật điều khiển với giao diện tương tác tự nhiên đang dần là một xu hướng chủ đạo trong các ứng dụng giải trí tương tác thân thiện.

Bên cạnh đó, các ứng dụng nhúng phát triển sử dụng FPGA (Field Programmable Gate Array) với những ưu điểm vượt trội như tốc độ xử lý nhanh, khả năng xử lý song song, tiêu hao năng lượng thấp, hiệu suất xử lý cao và giá thành thấp đang dần chiếm ưu thế trong thiết kế các ứng dụng thời gian thực, cho các ứng dụng trên thiết bị sử dụng pin so với các giải pháp phần mềm hoặc các bộ xử lý tín hiệu số chuyên dụng. Hơn nữa việc phát triển các ứng dụng nhúng trên nền FPGA cho phép

giảm thời gian, chi phí cho kiểm thử để nhanh chóng chuyển các ý tưởng thiết kế thành sản phẩm thương mại.

Những năm gần đây, đã có một nghiên cứu theo hướng thiết kế hệ thống theo dõi sự di chuyển của bàn tay (hand tracking) hoặc cử động của bàn tay (hand gestures) cho các ứng dụng điều khiển (Ahmad Athif Mohd Faudzi, *et al.*, 2012), (Amit Gupta, *et al.*, 2012), (Guangdong Liu, Zhongke Shi, 2011), (Hanting Lu, Kedari Eley, 2013). Các nghiên cứu trên đã sử dụng ngôn ngữ Verilog HDL để thiết kế phần cứng của hệ thống và áp dụng giải pháp nhận dạng màu da để nhận dạng và theo dõi sự di chuyển của bàn tay.

Trong nghiên cứu này, chúng tôi đã chọn phương án thiết kế phần cứng của hệ thống nhúng trên FPGA bằng cách sử dụng lõi IP (Intellectual Property core) được nhà sản xuất cung cấp. Các lõi IP này đã được tối ưu hóa nhằm đảm bảo cho thiết kế phần cứng của hệ thống nhúng hoạt động ổn định và hiệu quả. Ngôn ngữ C được sử dụng cho việc lập trình cài đặt nhanh các thuật toán xử lý ảnh nhằm theo dõi sự di chuyển của bàn tay trong thời gian thực phục vụ cho mục đích điều khiển thiết bị. Trong nghiên cứu này, một ứng dụng trò chơi đơn giản cũng được thực hiện với mục đích kiểm chứng tính năng điều khiển thiết bị của hệ thống được thiết kế. Nội dung của bài viết bao gồm: Phần 2 trình bày phương pháp thiết kế hệ thống nhúng, giải thuật phần mềm để phát hiện, xử lý ảnh và điều khiển, Phần 3 đánh giá kết quả thực nghiệm. Kết luận và kiến nghị được trình bày ở Phần 4.

2 THIẾT KẾ HỆ THỐNG NHÚNG

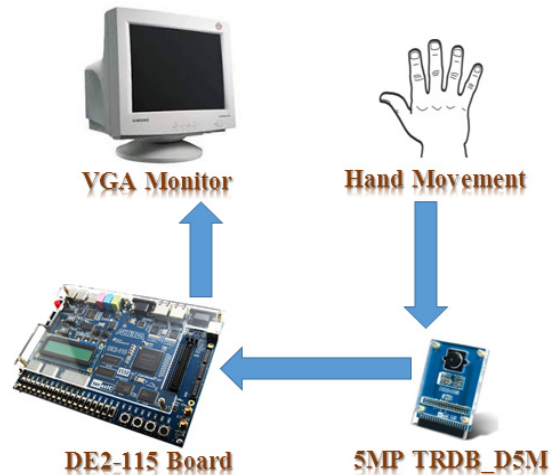
2.1 Tổng quan hệ thống

Hệ thống nhúng trong nghiên cứu này sử dụng Altera DE2-115 với các ngoại vi chính bao gồm SDRAM và SRAM, camera 5MP TRDB_D5M và màn hình hiển thị như Hình 1.

– *Board DE2-115*: được sử dụng để xây dựng hệ thống nhúng cho phép giao tiếp với camera chụp và xử lý hình ảnh bàn tay để điều khiển, hiển

thị trò chơi trên màn hình VGA.

- *Camera 5MP TRDB_D5M*: thu nhận hình ảnh từ bàn tay và gửi dữ liệu về hệ thống nhúng.
- *VGA Monitor*: màn hình theo chuẩn VGA để hiển thị trò chơi.
- *Bàn tay*: sự di chuyển của bàn tay cho phép điều khiển trò chơi đã được thiết kế sẵn.

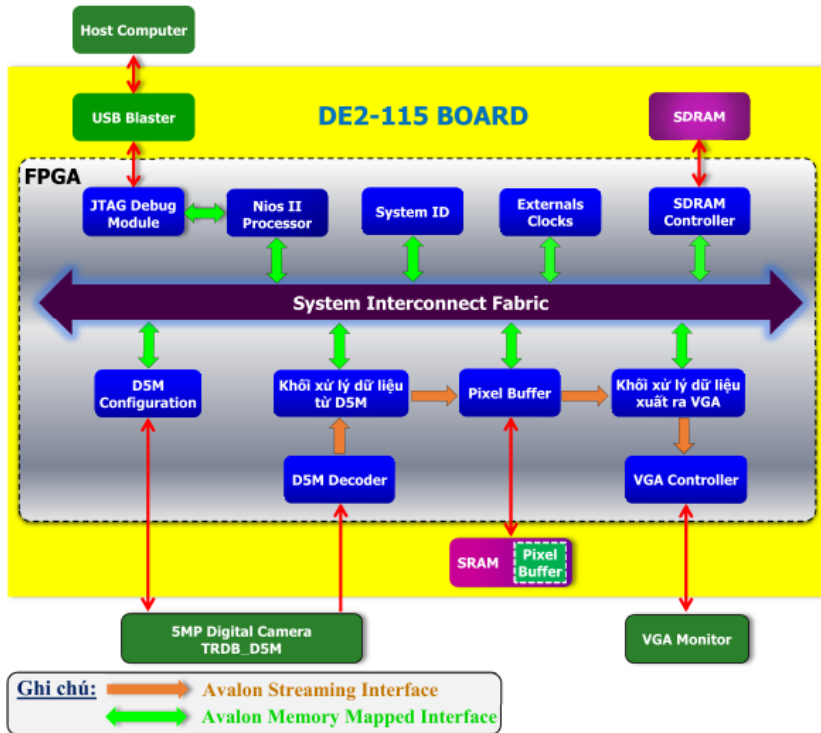


Hình 1: Sơ đồ hệ thống

2.2 Xây dựng phần cứng của hệ thống nhúng

Phần cứng của hệ thống được xây dựng trên chip FPGA Cyclone IV của board Altera DE2-115 bằng cách sử dụng các lõi IP thích hợp với trong công cụ SOPC Builder của phần mềm Altera Quartus II.

Mỗi khối trong sơ đồ trên sẽ tương ứng với một lõi IP trong hệ thống SOPC. Các lõi IP này giao tiếp với nhau dựa trên kết nối liên hệ thống (System Interconnect Fabric) có tốc độ cao và băng thông rộng. Kết nối này sử dụng ghép nối ảnh xạ bộ nhớ (Avalon Memory-Mapped Interface) và ghép nối luồng dữ liệu (Avalon Streaming Interface) để thực hiện việc truyền nhận dữ liệu chính xác giữa các thành phần trong hệ thống (Altera Corporation, 2010).



Hình 2: Sơ đồ khối hệ thống

2.2.1 Khối xử lý trung tâm

Khối xử lý trung tâm sử dụng CPU Nios II/ hoạt động ở tần số 50 MHz có bộ nhớ chương trình là SDRAM với cache lệnh và cache dữ liệu lần lượt là 4 KBytes và 2 Kbytes (Altera Corporation, 2011).

2.2.2 Các khối thu nhận và xử lý dữ liệu từ camera TRDB_D5M

Gồm 3 khối chính như sau (Altera Corporation, 2010):

- DSM configuration có nhiệm vụ khởi tạo và cấu hình hoạt động cho 5MP digital Camera (TRDB_D5M).

- DSM Decoder có nhiệm vụ lấy dữ liệu từ nguồn video do camera cung cấp và truyền dữ liệu nhận được đến khối DSM Bayer Pattern.

- Khối xử lý dữ liệu TRDB_D5M có nhiệm vụ xử lý nguồn dữ liệu do khối DSM Decoder cung cấp.

2.2.3 Khối Pixel Buffer

Khối Pixel Buffer được là bộ phận rất quan trọng của hệ thống do được sử dụng trong cả hai hệ thống nhưng thành phần là hệ thống giao tiếp với camera và giao tiếp với lõi VGA. Pixel Buffer có 2

nhiệm vụ vừa bộ nhớ đệm cho dữ liệu thu được từ camera và xuất dữ liệu ra màn hình VGA.

2.2.4 Các khối xử lý dữ liệu VGA

Gồm 2 khối chính (Altera Corporation, 2010):

- Khối xử lý dữ liệu xuất ra VGA có nhiệm vụ đọc dữ liệu khung ảnh (frame) được lưu trữ trong bộ nhớ Pixel Buffer (vùng nhớ đệm dành cho VGA) và thực hiện chuyển đổi định dạng phù hợp với chuẩn hiển thị VGA. Khối này bao gồm các khối con sau:

Khối VGA Pixel Buffer đọc các dữ liệu được ghi trên bộ nhớ đệm Pixel Buffer và truyền dữ liệu này sang khối VGA RGB Resampler, nhằm mục đích hiển thị nội dung khung ảnh lên VGA.

Khối RGB Resampler có nhiệm vụ chuyển đổi định dạng 8-bit Grayscale (nhận được từ Pixel Buffer DMA Controller) sang RGB 30-bit RGB – chuẩn phù hợp cho hoạt động của khối VGA Controller.

Khối VGA Scaler có nhiệm vụ chuyển đổi độ phân giải của dòng dữ liệu video về định dạng phù hợp với dữ liệu ngõ vào của khối VGA Controller. Khối này nhận dữ liệu từ khối VGA RGB Resampler có độ phân giải là 80x60 và chuyển đổi

chúng sang 640x480 – độ phân giải phù hợp cho hoạt động của khối VGA Controller.

Khối VGA Dual-Clock FIFO đệm dữ liệu video và lưu trữ nó cho đến khi khối VGA Controller sẵn sàng để hiển thị. Bên cạnh đó, khối Dual-Clock FIFO còn có vai trò đảm bảo sự đồng bộ và giao tiếp giữa hệ thống với khối VGA controller.

– *Khối VGA Controller* tạo các tín hiệu điều khiển để hiển thị dữ liệu video lên màn hình VGA. Dữ liệu cho khối VGA Controller được cung cấp từ khối Dual-Clock FIFO theo định dạng 30-bit RGB với độ phân giải 640x480.

2.2.5 Các khối xử lý âm thanh

Để có thể xuất âm thanh trò chơi ra loa (speaker), dữ liệu âm thanh được đọc trực tiếp từ bộ nhớ Flash và xuất ra cổng Line Out của Audio (Altera Corporation, 2013). Để làm được điều này, ta cần các khối chức năng sau:

– *Audio Configuration* có nhiệm vụ cấu hình cho hoạt động của chip Audio Codec thông qua chuẩn giao tiếp I²C. Trong SOPC Builder, lõi IP được sử dụng là Audio/Video Configuration.

– *Audio* có nhiệm vụ xuất tín hiệu âm thanh ra cổng Line Out trên board DE2-115. Lõi IP tương ứng được sử dụng là Audio.

– *Flash* có nhiệm vụ đọc dữ liệu âm thanh đã được lưu trữ trước đó và nạp vào khối Audio để xuất âm thanh ra loa. File âm thanh phục vụ cho trò chơi được nạp trước đó thông qua phần mềm DE2-115 Control Panel. Lõi IP được sử dụng là Altera UP Flash Memory.

2.2.6 Các khối chức năng khác

Khối External Clocks có nhiệm vụ cung cấp xung đồng hồ cần thiết cho các thiết bị ngoại vi bao gồm SDRAM, chip on-board VGA DAC và cả nguồn xung đồng hồ cho camera 5MP TRDB_D5M. Hệ thống thiết kế với khối External Clocks sử dụng lõi IP Avalon ALTPLL để tạo

xung đồng hồ cho toàn hệ thống (Altera Corporation, 2011).

2.3 Thiết kế phần mềm

2.3.1 Chương trình chính

Chương trình chính có nhiệm vụ thực hiện tạo giao diện và điều khiển trò chơi bằng cách gọi các chương trình con thực hiện các chức năng liên quan: chụp và lưu ảnh, xử lý ảnh, hiển thị các giao diện và điều khiển trò chơi. Hình 3 trình bày lưu đồ mô tả cách hoạt động của chương trình này.

2.3.2 Chương trình con Đọc dữ liệu từ camera

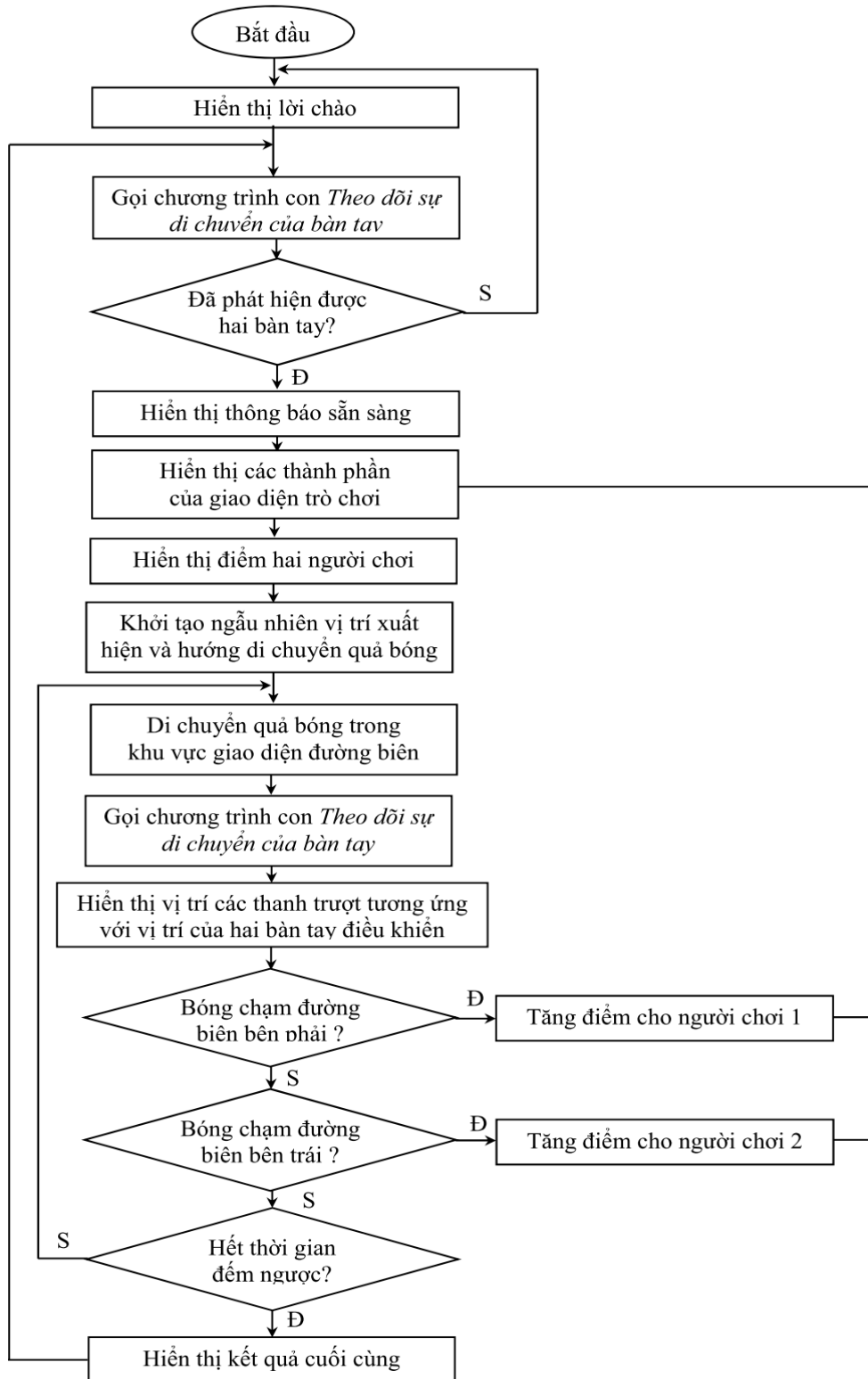
Hệ thống được thiết kế để nhận dữ liệu liên tục từ camera TRDB_D5M. Các khung ảnh được ghi liên tục lên một vùng nhớ của SRAM được chọn làm bộ nhớ đệm Pixel Buffer ở vị trí ghi. Dữ liệu từ camera được lưu tại vùng nhớ này để phục vụ cho các công việc xử lý.

2.3.3 Chương trình con Tách ảnh bàn tay

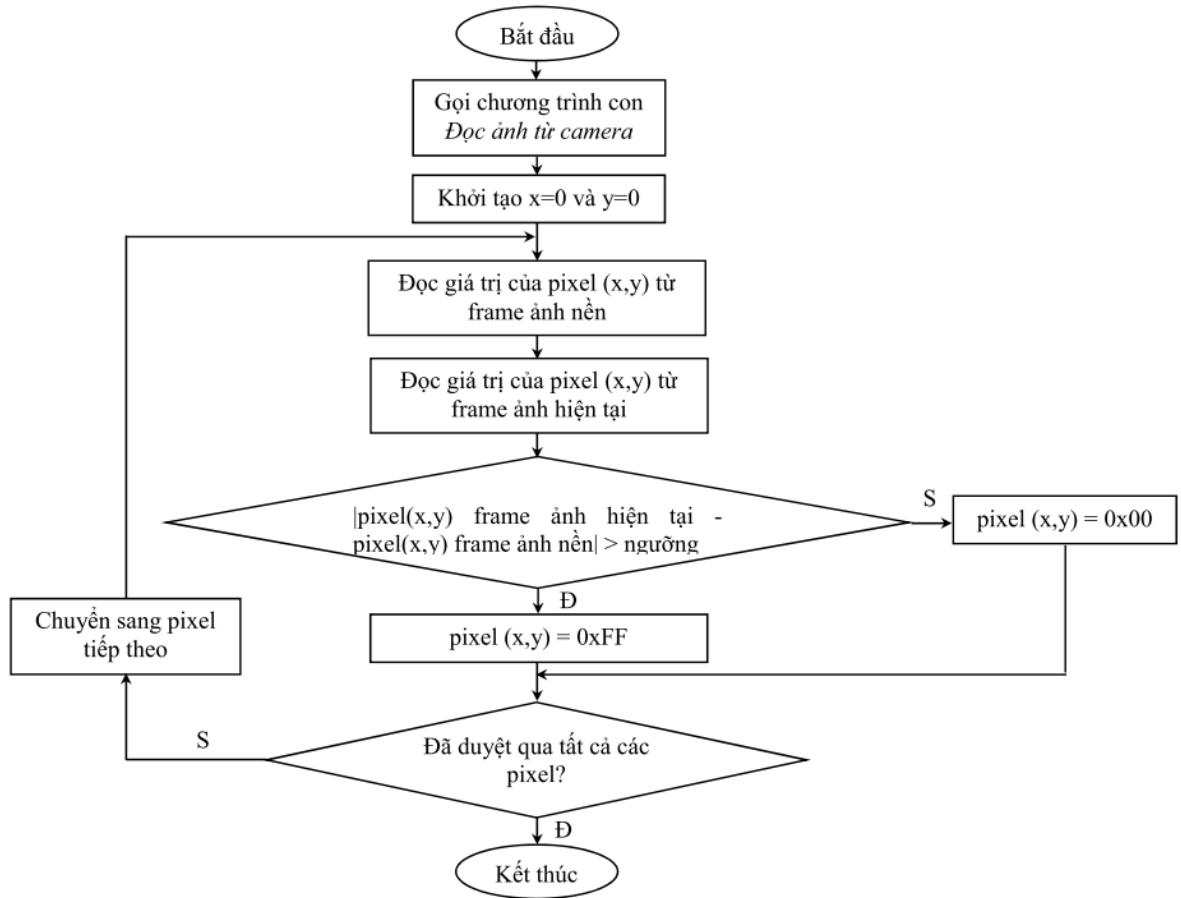
Để có thể tách ảnh bàn tay của người điều khiển (người chơi), nghiên cứu này sử dụng phương pháp trừ ảnh kết hợp với phân ngưỡng để cho ra ảnh mức xám chỉ bao gồm hai mức sáng là 0x00 và 0xFF. Lưu đồ của chương trình con này được trình bày chi tiết ở Hình 4 (Lâm Thị Ngọc Châu, 2003).

2.3.4 Chương trình con Theo dõi bàn tay

Trong thực tế, hình ảnh thu được từ camera sẽ xuất hiện rất nhiều nhiễu. Một bộ lọc thông thấp đã được sử dụng để hạn chế nhiễu trong, tuy nhiên nhiễu vẫn còn xuất hiện nhiều. Do đó, để theo dõi tốt quá trình di chuyển của bàn tay trong một phạm vi nhất định, chương trình sẽ thực hiện việc gán một cửa sổ có kích thước 3x3 pixel cho bàn tay như Hình 5(a). Việc xét 09 pixel kề nhau này tạo thành một khối hình vuông để xác định vị trí của bàn tay sẽ đảm bảo không bị nhầm lẫn với nhiễu đồng thời giúp cho việc theo dõi bàn tay khi di chuyển sẽ dễ dàng hơn mà không ảnh hưởng đến tốc độ xử lý.



Hình 3: Lưu đồ chương trình chính



Hình 4: Lưu đồ chương trình con Tách ảnh bàn tay

Do giới hạn về khả năng xử lý của CPU Nios II nên giải thuật trừ ảnh kết hợp với phân ngưỡng ảnh xám được chọn sử dụng để tách ảnh bàn tay nhằm đảm bảo khả năng xử trong thời gian thực của hệ thống. Kết quả của thu được sau khi áp dụng giải thuật này ta sẽ có ảnh xám của ảnh chụp từ camera

với các pixel có giá trị bằng 0xFF tương ứng với ảnh hai bàn tay. Ảnh xám này sau khi bỏ bớt phần nội dung ảnh không chứa ảnh bàn tay (nhằm giảm số pixel cần xử lý, góp phần tăng tốc độ chương trình) sẽ được lưu vào 02 mảng riêng biệt tương ứng với ảnh của từng bàn tay.

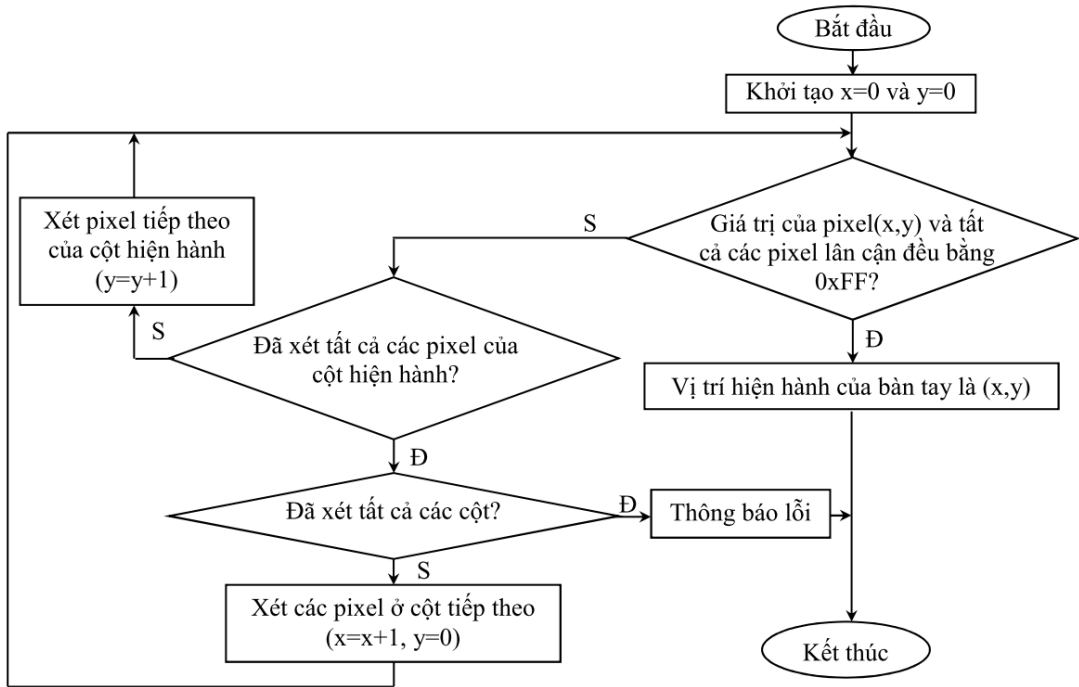


(a)



(b)

Hình 5: (a) Vị trí cửa sổ 3x3 pixel và (b) Hình ảnh theo dõi bàn tay trong thực tế



Hình 6: Lưu đồ giải thuật Theo dõi bàn tay

Để điều khiển hai thanh trượt trong ứng dụng trò chơi, chương trình con *Theo dõi sự di chuyển của bàn* sẽ áp dụng giải thuật *Theo dõi bàn tay* (xem lưu đồ Hình 6) để xử lý lần lượt với 02 mảng lưu ảnh xám của từng bàn tay có được từ chương trình con *Tách ảnh bàn tay*. Trong giải thuật này, các pixel trong mảng lưu ảnh xám của bàn tay được kiểm tra lần lượt ưu tiên theo từng dòng từ trên xuống dưới ($y=0..n$) và cột từ trái sang phải ($x=0..n$). Pixel(x,y) đầu tiên có giá trị 0xFF và tất cả các pixel lân cận của nó cũng có giá trị bằng

0xFF thì (x,y) được xem như vị trí của bàn tay. Giá trị y này sẽ được gán cho vị trí của con trượt tương ứng.

3 KẾT QUẢ THỰC NGHIỆM

Hình 7 là hình ảnh của thiết kế phần cứng của hệ thống trên nền FPGA sử dụng phần mềm Altera SOPC Builder.

Hệ thống phần cứng hoạt động ổn định, sử dụng ít tài nguyên trên chip FPGA (xem Hình 8).

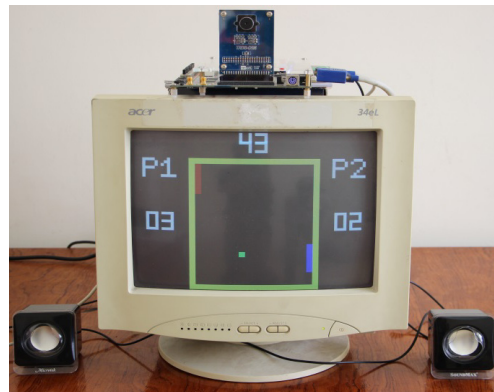
<input checked="" type="checkbox"/>	<input type="checkbox"/>	CPU	Nios II Processor	sys_clk	<input checked="" type="checkbox"/>	0x00200800	0x00200fff
<input checked="" type="checkbox"/>	<input type="checkbox"/>	SDRAM	SDRAM Controller	sys_clk	<input checked="" type="checkbox"/>	0x10000000	0x17ffff
<input checked="" type="checkbox"/>	<input type="checkbox"/>	sysid	System ID Peripheral	sys_clk	<input checked="" type="checkbox"/>	0x00201060	0x00201067
<input checked="" type="checkbox"/>	<input type="checkbox"/>	External_Clock	Clock Signals for DE-Series Board Peri...	clk	<input checked="" type="checkbox"/>	0x00201068	0x00201069
<input checked="" type="checkbox"/>	<input type="checkbox"/>	D5M_Config	Audio and Video Config	sys_clk	<input checked="" type="checkbox"/>	0x00201010	0x0020101f
<input checked="" type="checkbox"/>	<input type="checkbox"/>	D5M_Decoder	Video-In Decoder	sys_clk			
<input checked="" type="checkbox"/>	<input type="checkbox"/>	D5M_Bayer_Resampler	Bayer Pattern Resampler	sys_clk			
<input checked="" type="checkbox"/>	<input type="checkbox"/>	D5M_Clipper	Clipper	sys_clk			
<input checked="" type="checkbox"/>	<input type="checkbox"/>	D5M_Scaler	Scaler	sys_clk			
<input checked="" type="checkbox"/>	<input type="checkbox"/>	D5M_Scaler_2	Scaler	sys_clk			
<input checked="" type="checkbox"/>	<input type="checkbox"/>	D5M_RGB_Resampler	RGB Resampler	sys_clk			
<input checked="" type="checkbox"/>	<input type="checkbox"/>	D5M_DMA_Controller	DMA Controller	sys_clk	<input checked="" type="checkbox"/>	0x00201020	0x0020102f
<input checked="" type="checkbox"/>	<input type="checkbox"/>	Pixel_Buffer	SRAM/SSRAM Controller	sys_clk	<input checked="" type="checkbox"/>	0x00000000	0x001ffff
<input checked="" type="checkbox"/>	<input type="checkbox"/>	VGA_Pixel_Buffer	Pixel Buffer DMA Controller	sys_clk	<input checked="" type="checkbox"/>	0x00201000	0x0020100f
<input checked="" type="checkbox"/>	<input type="checkbox"/>	VGA_RGB_Resampler	RGB Resampler	sys_clk			
<input checked="" type="checkbox"/>	<input type="checkbox"/>	VGA_Scaler	Scaler	sys_clk			
<input checked="" type="checkbox"/>	<input type="checkbox"/>	VGA_Dual_Clock_FIFO	Dual-Clock FIFO	multiple			
<input checked="" type="checkbox"/>	<input type="checkbox"/>	VGA_Controller	VGA Controller	vga_clk			
<input checked="" type="checkbox"/>	<input type="checkbox"/>	Hex3_Hex0	Parallel Port	sys_clk	<input checked="" type="checkbox"/>	0x00201030	0x0020103f
<input checked="" type="checkbox"/>	<input type="checkbox"/>	Hex7_Hex4	Parallel Port	sys_clk	<input checked="" type="checkbox"/>	0x00201040	0x0020104f
<input checked="" type="checkbox"/>	<input type="checkbox"/>	Switches	Parallel Port	sys_clk	<input checked="" type="checkbox"/>	0x00201050	0x0020105f

Hình 7: Hệ thống nhúng được thiết kế trên FPGA

Flow Summary	
Flow Status	Successful - Sun May 18 01:36:39 2014
Quartus II Version	10.1 Build 153 11/29/2010 SJ Full Version
Revision Name	PINGPONG
Top-level Entity Name	PINGPONG
Family	Cyclone IV E
Device	EP4CE115F29C7
Timing Models	Final
Total logic elements	8,837 / 114,480 (8 %)
Total combinational functions	6,753 / 114,480 (6 %)
Dedicated logic registers	6,425 / 114,480 (6 %)
Total registers	6544
Total pins	270 / 529 (51 %)
Total virtual pins	0
Total memory bits	88,432 / 3,981,312 (2 %)
Embedded Multiplier 9-bit elements	4 / 532 (< 1 %)
Total PLLs	1 / 4 (25 %)

Hình 8: Tài nguyên trên chip FPGA được sử dụng thiết kế phần cứng của hệ thống

Bố trí hệ thống trong thực nghiệm được trình bày ở Hình 9. Hệ thống hoạt động ổn định thực hiện việc phát hiện và theo dõi sự di chuyển của 02 bàn tay của người chơi đứng đối diện với camera với các khoảng cách cần được đảm bảo như Hình 10. Trong ứng dụng trò chơi được thiết kế thì sự di chuyển của bàn tay của từng người chơi sẽ điều



Hình 9: Bố trí của hệ thống trong thực nghiệm

khiển được thanh trượt tương ứng. Giao diện trò chơi này được thiết kế đơn giản, màu sắc dễ nhìn kết hợp hiệu ứng âm thanh khi chơi tạo thêm hứng thú cho người. Thực nghiệm trong nhiều môi trường có độ chiếu sáng khác nhau cho thấy hầu hết đều có thể hoạt động tốt khi điều kiện ánh sáng xung quanh không thay đổi đột ngột.



Hình 10: Các vị trí và khoảng cách người chơi nên đảm bảo

Tuy nhiên, do hệ thống hiện sử dụng camera đi chung với board Altera DE2 có tốc độ khung hình không cao (15 fps) làm ảnh hưởng đến tốc độ phát hiện, theo dõi sự di chuyển của bàn tay đưa đến tốc độ điều khiển còn hạn chế. Mặt khác, phương pháp trừ ảnh mặc dù cho tốt độ xử lý cao hơn phương pháp nhận dạng màu da nhưng sẽ bị ảnh hưởng nhiều bởi điều kiện chiếu sáng của môi trường, vị trí các vật xung quanh thay đổi. Một đoạn video clip trình bày kết quả của nghiên cứu có thể xem tại đường link <http://goo.gl/orX0xB>.

4 KẾT LUẬN

Bài viết đã trình bày việc thiết kế một bộ điều khiển dựa vào sự di chuyển của bàn tay trên nền FPGA, được kiểm chứng thành công qua một ứng dụng trò chơi trực quan, sinh động. Các thực nghiệm trong các môi trường có điều kiện chiếu sáng khác nhau, hệ thống đều hoạt động tốt, đạt các yêu cầu đặt ra. Kết quả của nghiên cứu cho thấy hệ thống điều khiển dựa trên sự di chuyển của bàn tay

hoàn toàn có thể áp dụng cho các ứng dụng điều khiển tương tác, thân thiện với người sử dụng.

TÀI LIỆU THAM KHẢO

- Ahmad Athif Mohd Faudzi, Muaammar Hadi Kuzman Ali, M. Asyraf Azman, Zool Hilmi Ismail. Real-time Hand Gestures System for Mobile Robots Control. ScienDirect, Procedia Engineering 41, 2012, pp. 798-804.
- Altera Corporation, 2007. Quartus II Handbook Volume 4: SOPC Bulder. URL: www.cs.columbia.edu/~sedwards/classes/2008/4840/qts_qii5v4.pdf, ngày truy cập 15/09/2014.
- Altera Corporation, 2010. Media Computer System for the Altera DE2-115 Board. URL: [ftp://ftp.altera.com/up/pub/Altera.../DE2-115_Media_Computer.pdf](http://ftp.altera.com/up/pub/Altera.../DE2-115_Media_Computer.pdf), ngày truy cập 15/09/2014.
- Altera Corporation, 2010. Video IP Cores for Altera DE Series Boards. URL: [ftp://ftp.altera.com/up/pub/Altera_Material/9.1/University_Program_IP_Cores/Audio_Video/Video.pdf](http://ftp.altera.com/up/pub/Altera_Material/9.1/University_Program_IP_Cores/Audio_Video/Video.pdf), ngày truy cập 15/09/2014.
- Altera Corporation, 2011. Nios II Processor Reference Handbook. URL: www.altera.com/literature/hb/nios2/n2cpu_nii5v1.pdf, ngày truy cập 15/09/2014.
- Altera Corporation, 2011. Nios II Software Developer's Handbook. 2011. URL: www.altera.com/literature/hb/nios2/n2sw_nii5v2.pdf, ngày truy cập 15/09/2014.
- Altera Corporation, 2013. University Program. Computer Organization - IP Cores. URL: http://www.altera.com/education/univ/materials/comp_org/ip-cores/unv-ip-cores.html, ngày truy cập 15/09/2014.
- Amit Gupta, Vijay Kumar Sehrawat, Mamta Khosla. FPGA Based Real Time Human Gesture Recognition System. Procedia Technology 6, 2012, pp. 98-107.
- Chao-Tang Li, Wen-Hui Chen. A Novel FPGA-based Hand Gesture Recognition System. Journal of Convergence Information technology (JCIT), Volume 7, Number 9, May 2012, pp. 22-229.
- Guangdong Liu, Zhongke Shi, 2011. Embedded implementation of real-time skin detection system. 2011 International Conference on Transportation, Mechanical, and Electrical Engineering (TMEE), December 16-18. Changchun, China, pp. 2463-2466.
- Hanting Lu, Kedari Eley, 2013. Hand Tracking Pong. URL: http://people.ece.cornell.edu/land/courses/ece5760/FinalProjects/s2013/hl855_kre27/hl855_kre27/index.html, ngày truy cập 15/09/2014.
- Lâm Thị Ngọc Châu, 2003. Kỹ thuật đồ họa. Khoa CNTT&TT. Đại học Cần Thơ. 159 trang.
- Rumaisa Iftikhar, Samiya, Hira Abbasi, Tahir Quadri, Sabia Abbasi and Naeem Abbas. Controlling of Mouse Cursor using Video Processing in FPGA. International Journal of Current Engineering and Technology. Vol. 4, No. 2, April 2014, pp. 612-615.
- Stefen Oniga, Janos Vegh, Ioan Orha, 2012. Intelligent Human-Machine Interface Using Hand Gestures Recognition. Automation Quality and Testing Robotics (AQTR), 2012 IEEE International Conference on 24-27 May 2012. Cluj-Napoca, Rumania, pp. 559-563.
- Terasic Technologies Inc, 2009. TRDB-D5M Hardware Specification. URL: https://courses.cs.washington.edu/courses/cs467/08au/labs/Resources/THDB-D5M_Hardware%20specification.pdf, ngày truy cập 15/09/2014.
- Terasic Technologies Inc, 2013. DE2-115 User Manual. URL: http://www.terasic.com.tw/cgi-bin/page/archive_download.pl?Language=English&No=502&FID=cd9c7c1feaa2467c58c9aa4cc02131af, ngày truy cập 15/09/2014.
- Terasic Technologies Inc, 2013. TRDB_D5M User Guide. URL: https://www.terasic.com.tw/attachment/archive/281/TRDB_D5M_UserGuide.pdf, ngày truy cập 15/09/2014.