BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT
THÀNH PHỐ HỒ CHÍ MINH

ĐỒ ÁN TỐT NGHIỆP
NGÀNH CÔNG NGHỆ KỸ THUẬT MÁY TÍNH

XÂY DỰNG HỆ THỐNG SOC CÔNG SUẤT THẤP TRÊN FPGA VÀ ỨNG DỤNG TRÒ CHƠI PING-PONG

GVHD: VÕ MINH HUÂN
SVTH : SẦM BẾ TRƯỜNG
MSSV: 11119142

SKL005311

Tp. Hồ Chí Minh, tháng 06/2016

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH
KHOA ĐIỆN ĐIỆN TỬ
BỘ MÔN KỸ THUẬT MÁY TÍNH - VIỄN THÔNG

ĐỒ ÁN TỐT NGHIỆP

XÂY DỰNG HỆ THỐNG SoC CÔNG SUẤT THẤP TRÊN FPGA VÀ ỨNG DỤNG TRÒ CHƠI PING-PONG

NGÀNH CÔNG NGHỆ KỸ THUẬT MÁY TÍNH

Sinh viên: **SẦM BẾ TRƯỜNG**
MSSV: 11119142

TP. HỒ CHÍ MINH – 6/2016

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT THÀNH PHỐ HỒ CHÍ MINH
KHOA ĐIỆN ĐIỆN TỬ
BỘ MÔN KỸ THUẬT MÁY TÍNH - VIỄN THÔNG

ĐỒ ÁN TỐT NGHIỆP

XÂY DỰNG HỆ THỐNG SoC CÔNG SUẤT THẤP TRÊN FPGA VÀ ỨNG DỤNG TRÒ CHƠI PING-PONG

NGÀNH CÔNG NGHỆ KỸ THUẬT MÁY TÍNH

Sinh viên: **SẦM BẾ TRƯỜNG**
MSSV: 11119142

Hướng dẫn**: TS. VÕ MINH HUÂN**

TP. HỒ CHÍ MINH – 6/2016

LỜI CẢM ƠN

Lời đầu tiên, người thực hiện đề tài xin được phép chân thành gửi lời cảm ơn đến thầy Võ Minh Huân, giáo viên hướng dẫn đề tài đã định hướng và trao đổi những kinh nghiệm quý báu để hoàn thành những nội dung trong đề tài một cách hoàn chỉnh.

Nhóm thực hiện đề tài cũng xin trân trọng cảm ơn các thầy (cô) trong trường Đại Học Sư Phạm Kỹ Thuật THCM nói chung và thầy (cô) khoa Điện – Điện Tử đã tạo điều kiện và hỗ trợ các trang thiết bị để người thực hiện đề tài hoàn thành tốt đề tài.

Cũng gửi lời cảm ơn đến gia đình: bố mẹ, anh chị em đã tạo điều kiện thuận lợi về mặt vật chất và tinh thần là cơ sở vững chắc về tâm lý để người thực hiện đề tài hoàn thành tốt công việc học tập của mình.

TPHCM, Ngày Tháng 6 Năm 2016

Người Thực Hiện

Sầm Bế Trường

TÓM TẮT LUẬN VĂN

FPGA đang là một lựa chọn hàng đầu trong việc xây dựng các hệ thống nhằm mục đích nghiên cứu, hoặc xây dựng các ứng dụng. Trong luận văn này chúng tôi tiến hành xây dựng một hệ thống tích hợp trên một chip FPGA sử dụng board FPGA Altera DE2-115 và các thiết bị ngoại vi như Loa, thẻ nhớ SD, màn hình LCD, bàn phím máy tính PS/2. Bên cạnh đó chúng tôi xây dựng một ứng dụng sử dụng ngôn ngữ lập trình C nhằm xây dựng một ứng dụng nhằm mục đích giải trí cho người dùng hệ thống như chơi trò chơi, nghe nhạc. Ngoài ra chúng tôi còn tối ưu năng lượng cho hệ thống bằng việc xây dựng chế độ nghỉ cho hệ thống khi người dùng không dùng đến hệ thống hoặc do người dùng lựa chọn chế độ nghỉ.

MỤC LỤC

DANH MỤC HÌNH IX
DANH MỤC BẢNG XI
CÁC TỪ VIẾT TẮT XII

CHƯƠNG 1 GIỚI THIỆU **1**

1.1 GIỚI THIỆU 1
1.2 SƠ ĐỒ HỆ THỐNG 1
1.3 PHẠM VI ÁP DỤNG 2

CHƯƠNG 2 CƠ SỞ LÝ THUYẾT **3**

2.1 TỔNG QUAN VỀ HỆ THỐNG NHÚNG TRÊN FPGA 3
2.1.1 Giới thiệu *4*
2.1.2 Phần cứng kit FPGA Altera DE2-115 ..... *5*
2.2 NGÔN NGỮ MÔ TẢ PHẦN CỨNG VERILOG VÀ HỆ THỐNG SOC 6
2.2.1 Ngôn ngữ mô tả phần cứng Verilog *7*
2.2.2 Hệ thống System on Chip (SoC) *8*
2.3 PHƯƠNG PHÁP GIẢM NĂNG LƯỢNG CLOCK GATING 9
2.3.1 Tiết kiệm năng lượng trong thiết kế ASIC *9*
2.3.2 Giải pháp tiết kiệm năng lượng clock gating *9*
2.4 GIỚI THIỆU CÁC PHẦN MỀM HỖ TRỢ 11
2.4.1 Quartus II *11*
2.4.2 Nios II Embedded Design Suite *12*

CHƯƠNG 3 XÂY DỰNG HỆ THỐNG **14**

3.1 XÂY DỰNG PHẦN CỨNG 14
3.1.1 Sơ đồ hệ thống phần cứng *14*
3.1.2 Thiết kế module giao tiếp với bàn phím chuẩn PS/2 ..... *15*
3.1.3 Thiết kế hệ thống SoC *17*
3.1.4 Thiết kế hệ thống clock gating *26*
3.2 XÂY DỰNG PHẦN MỀM 29
3.2.1 Sơ đồ hệ thống chương trình *29*

3.2.2 Ứng dụng trò chơi ping-pong.....31
3.2.3 Sơ đồ hoạt động của các thiết bị ngoại vi 33
CHƯƠNG 4 KẾT QUẢ THỰC HIỆN 38
4.1 KẾT QUẢ XÂY DỰNG HỆ THỐNG.....38
4.2 KẾT QUẢ KHI DÙNG CLOCK GATING CHO HỆ THỐNG.....42
CHƯƠNG 5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....47
5.1 KẾT LUẬN.....47
5.2 HƯỚNG PHÁT TRIỂN DỀ TÀI 47
TÀI LIỆU THAM KHẢO 49

DANH MỤC HÌNH

Hình 1. 1: Sơ đồ hệ thống tổng quan đề tài 2

Hình 2. 1: Lưu đồ phát triển hệ thống nhúng trên FPGA 3
Hình 2. 2: Kit FPGA Altera DE2-115 5
Hình 2. 3: Minh họa ứng dụng ngôn ngữ verilog 7
Hình 2. 4: Mô tả các thành phần một SoC 8
Hình 2. 5: Giản đồ định thời khối clock gating 10
Hình 2. 6: Giao diện làm việc phần mềm Quartus II 11
Hình 2. 7: Giao diện làm việc trên Qsys 12
Hình 2. 8: Giao diện Nios II Embedded Design Suite 13

Hình 3. 1: Mô hình hệ thống phần cứng trên kit DE2-115. 14
Hình 3. 2: Sơ đồ khối module keyboard_controller 15
Hình 3. 3: Mô tả hoạt động module keyboard_controller bằng trạng thái máy 16
Hình 3. 4: Giản đồ định thời khung liệu truyền từ bàn phím 16
Hình 3. 5: Dạng sóng các tín hiệu mô phỏng bằng ModelSim 17
Hình 3. 6: Hệ thống SoC được xây dựng trên công cụ Qsys 17
Hình 3. 7: Sơ đồ khối vi xử lý Nios II 18
Hình 3. 8: Sơ đồ kết nối IP Jtag Uart 19
Hình 3. 9: Sơ đồ kết nối IP SDRAM controller và chip SDRAM 19
Hình 3. 10: Sơ đồ kết nối giữa IP pixel buffer và chip SRAM 20
Hình 3. 11: Sơ đồ kết nối giữa IP SD Card Controller và SD Card 20
Hình 3. 12: Sơ đồ kết nối giữa IP PS/2 và bàn phímPS/2 20
Hình 3. 13: Sơ đồ cấu hình cho hệ thống Audio 21
Hình 3. 14: Sơ đồ kết nối giữa VGA và hệ thống SoC 22
Hình 3. 15: Sơ đồ khối IP character buffer 22
Hình 3. 16: Sơ đồ khối IP pixel buffer dma 23
Hình 3. 17: Sơ đồ khối IP RGB resample 23

Hình 3. 18: Sơ đồ khối IP Scaler 24
Hình 3. 19: Sơ đồ khối IP Alpha Blender 24
Hình 3. 20: Sơ đồ khối IP Dual Clock FIFO 25
Hình 3. 21: Bộ ổn định xung clock 50Mhz cho hệ thống 25
Hình 3. 22: Bộ chia clock dùng cho vga 26
Hình 3. 23: Bộ chia xung clock cho chip SDRAM 26
Hình 3. 24: Bộ chia xung clock cho hệ thống Audio 26
Hình 3. 25: Sơ đồ thiết kế clock gating cho hệ thống SoC 27
Hình 3. 26: Sơ đồ bên trong khối clock gating 27
Hình 3. 27: Sơ đồ hệ thống trước (a) và sau (b) khi thiết kế clock gate 28
Hình 3. 28: Sơ đồ chức năng khối control_clock_gate 28
Hình 3. 29: Sơ đồ hoạt động của phần mềm 30
Hình 3. 30: Minh họa trò chơi ping-pong trên màn hình LCD 31
Hình 3. 31: Lưu đồ thuật toán trò chơi ping-pong 32
Hình 3. 32: Lưu đồ chương trình kết nối với bàn phím PS/2 33
Hình 3. 33: Lưu đồ chương trình phục vụ ngắt bàn phím 34
Hình 3. 34: Lưu đồ chương trình kết nối SD Card 35
Hình 3. 35: Lưu đồ chương trình kết nối audio 36
Hình 3. 36: Lưu đồ chương trình kết nối các phần cứng VGA 37

Hình 4. 1: Hệ thống nhúng thiết kế sử dụng Kit Altera DE2-115 38
Hình 4. 2: Sơ đồ phần cứng ở mức netlist sau khi tổng hợp 39
Hình 4. 3: Hệ thống thiết kế trên công cụ Qsys 39
Hình 4. 4: Lập trình phần mềm cho hệ thống phần cứng 41
Hình 4. 5: Thao tác với phần mềm hoạt động trên phần cứng được thiết kế 41
Hình 4. 6: (a), (b), (c), (d) Các thao tác với ứng dụng trò chơi ping-pong 42
Hình 4. 7: So sánh mức năng lượng tiêu tốn giữa các thiết kế 44
Hình 4. 8: So sánh công suất tiêu thụ động giữa các thiết kế 44
Hình 4. 9: Công suất tiêu thụ tĩnh giữa các thiết kế 45
Hình 4. 10: Công suất tiêu thụ các thành phần I/O 46

DANH MỤC BẢNG

Bảng 1: Bảng mô tả chức năng các tín hiệu module keyboard_controller 15
Bảng 2: Bảng các thông số phần cứng hệ thống sau khi tổng hợp 40
Bảng 3: So sánh các thông số tài nguyên phần cứng khi thiết kế..... 43

CÁC TỪ VIẾT TẮT

SoC	System on Chip
FPGA	Field-Programmable Gate Array
IP	Intellectual Property
SD	Secure Digital
HDL	Hardware Design Language
VHDL	Very High-level Design Language
PS/2	Personal System/2
DMA	Direct Memory Access
FAT	File Allocation Table
LEs	Logic Elements

Chương 1

GIỚI THIỆU

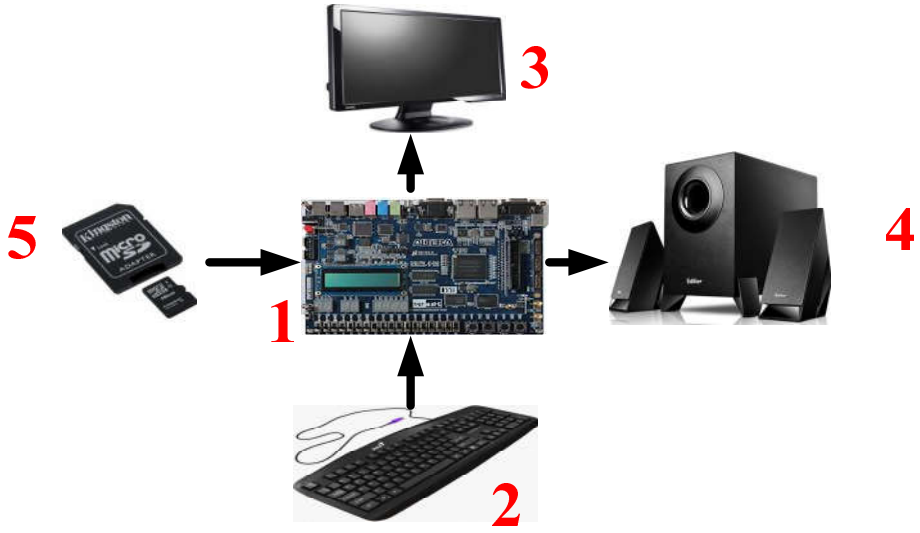
1.1 GIỚI THIỆU

Ngày nay các hệ thống nhúng sử dụng các con chip, vi điều khiển đã không còn quá xa lạ với con người hiện nay, đặc biệt là các nhà lập trình các hệ thống nhúng.

Hiện nay việc thiết kế hệ thống nhúng System-on-Chip [1] trên nền FPGA với nhiều ưu điểm vượt trội như tốc độ xử lý cao, tiêu tốn năng lượng thấp đang nhận được nhiều sự quan tâm nghiên cứu trong nhiều lĩnh vực và ứng dụng. Khác với các hệ thống nhúng thông thường, FPGA đáp ứng các nghiên cứu, ứng dụng một cách nhanh chóng mà không đòi hỏi thêm nhiều các thành phần ngoại vi cao để xử lý các ứng dụng khác nhau. Chính vì vậy, đã có nhiều đề tài nghiên cứu về xây dựng một hệ thống tích hợp trên một chip duy nhất. Bên cạnh đó, các hệ thống nhúng sử dụng SoC trên FPGA [2] cũng đang được ưa chuộng và phát triển mạnh mẽ trong thời gian gần đây dựa trên sự linh hoạt trong việc thay đổi thiết kế từ người dùng..

Từ thực tế trên, đề tài nghiên cứu "**Xây dựng hệ thống SoC công suất thấp trên FPGA và ứng dụng trò chơi Ping-Pong**" được thực hiện trên board DE2-115 với sự hỗ trợ của bộ công cụ Quartus và một số thiết bị ngoại vi như màn hình LCD, SD Card, loa, bàn phím PS/2, máy tính cá nhân nhằm nghiên cứu và phát triển một hệ thống nhúng phục vụ giải trí.

1.2 SƠ ĐỒ HỆ THỐNG



Hình 1. 1: Sơ đồ hệ thống tổng quan đề tài

Hệ thống bao gồm 5 thành phần chính: thẻ nhớ SD, loa, bàn phím, màn hình LCD và kit DE2-115. Đầu tiên, hệ thống SoC trên kit FPGA Altera DE2-115(1) sẽ thực hiện thiết lập giao tiếp với các ngoại vi: bàn phím chuẩn PS/2(2), màn hình LCD(3) kết nối thông qua cổng VGA, loa(4) kết nối thông qua cổng line-audio và SD Card(5).

Ứng dụng game Ping-Pong [3, 4] sẽ được hiển thị trên màn hình LCD, ứng dụng này được điều khiển bằng bàn phím chuẩn PS/2. Tùy thuộc vào chương trình trên hệ thống, hệ thống có thể đọc dữ liệu là các file nhạc dưới định dạng file là ".wav" để phát ra loa thông qua line-out Audio.

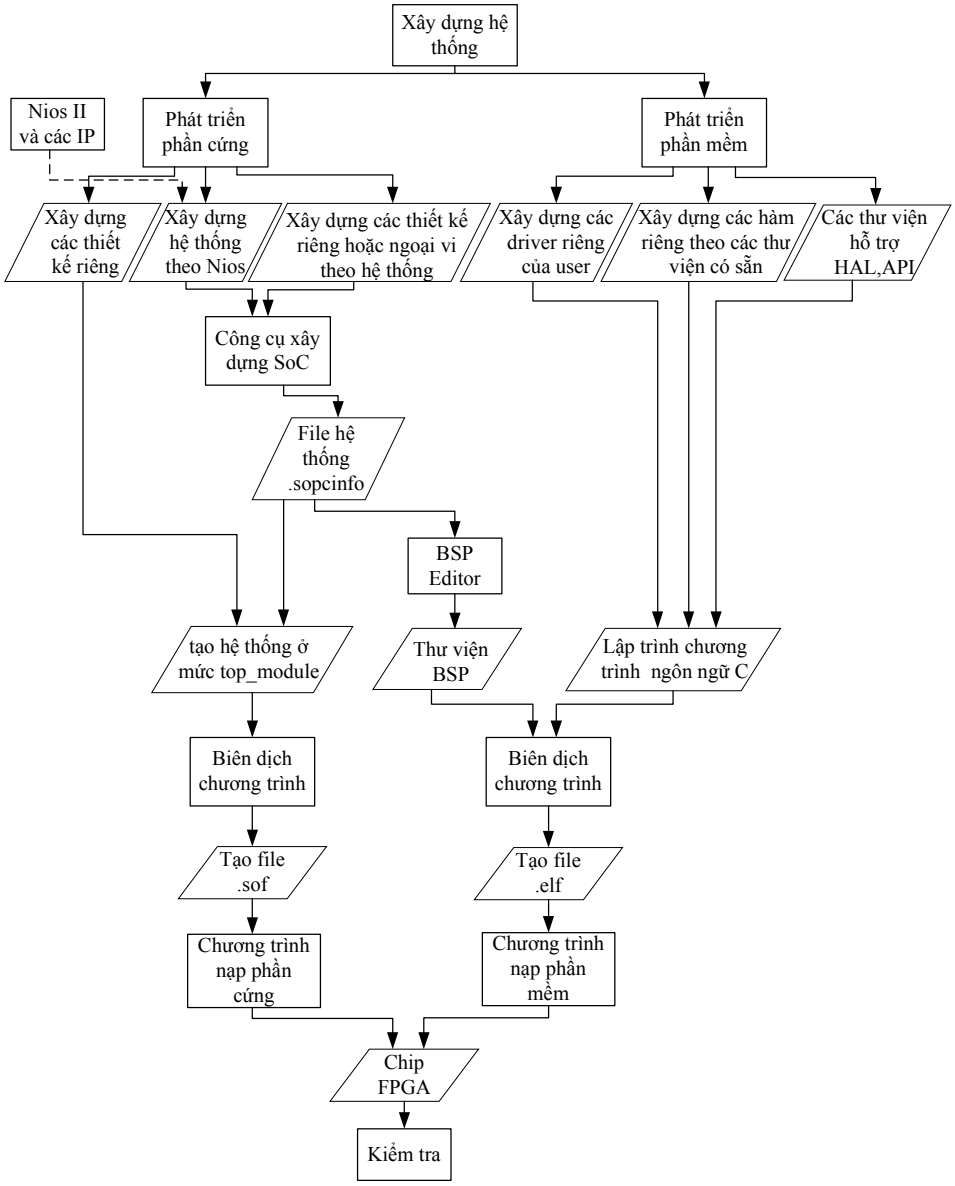
1.3 PHẠM VI ÁP DỤNG

Hệ thống nhúng được phát triển với ứng dụng trò chơi đơn giản được gọi là ping-pong được tương tác với người dùng thông qua màn hình LCD và điều khiển trò chơi bằng bàn phím chuẩn PS2. Ngoài ra hệ thống còn đáp ứng người dùng nghe nhạc với các định dạng file nhạc với địng dạng là .wav được lưu trữ trên thẻ nhớ SD định dạng FAT32 và chế độ nghỉ cho hệ thống khi người dùng lựa chọn chế độ nghỉ như các chế độ ngủ đông trên các máy tính cá nhân hiện nay nhằm tiết kiệm năng lượng cho hệ thống khi không sử dụng.

Chương 2

CƠ SỞ LÝ THUYẾT

2.1 TỔNG QUAN VỀ HỆ THỐNG NHÚNG TRÊN FPGA



Hình 2. 1: Lưu đồ phát triển hệ thống nhúng trên FPGA

2.1.1 Giới thiệu

FPGA (Field programmable Gate Array) là một loại mạch tích hợp cỡ lớn dùng cấu trúc mảng phần tử logic mà người dùng có thể lập trình được. Vi mạch FPGA được cấu thành từ các bộ phận như: các khối logic cơ bản lập trình được, hệ thống mạch liên kết, các khối vào/ra và một số thành phần khác như RAM, ROM. Thiết kế hay lập trình cho FPGA được thực hiện chủ yếu bằng các ngôn ngữ mô tả phần cứng HDL như VHDL, Verilog, AHDL, các hãng sản xuất FPGA lớn như Xilinx, Altera thường cung cấp các gói phần mềm và thiết bị phụ trợ cho quá trình thiết kế, cũng có một số các hãng thứ ba cung cấp các gói phần mềm kiểu này như Synopsys, Synplify.

Một số ứng dụng của FPGA như: Thiết kế hệ thống xử lý tín hiệu số DSP, tiền thiết kế mẫu ASIC, các hệ thống giao tiếp, điều khiển, nhận dạng ảnh. FPGA có tính linh động cao nên có thể đáp ứng nhiều ứng dụng

Quá trình xây dựng một hệ thống nhúng SoC [2] trên FPGA bao gồm bốn quá trình cơ bản là: kế hoạch xây dựng hệ thống, xây dựng phần cứng, xây dựng phần mềm, kiểm tra hệ thống.

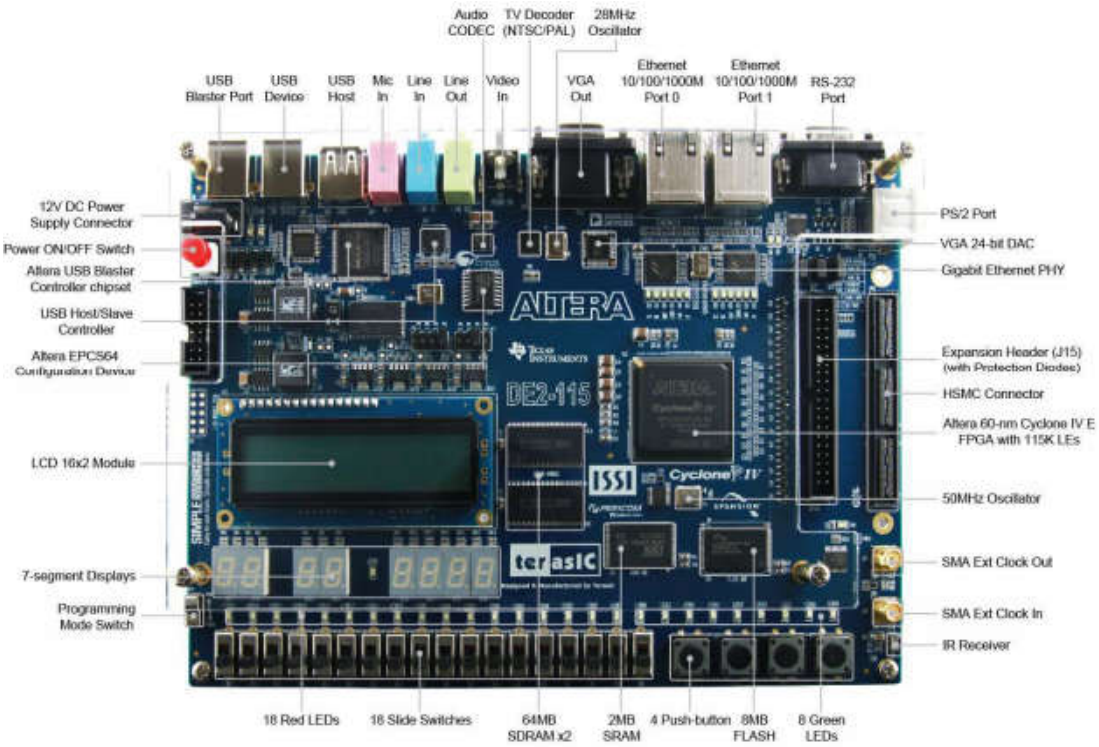
- *Quá trình lên kế hoạch xây dựng hệ thống*: quá trình bắt đầu khi có các yêu cầu về xây dựng hệ thống SoC. Người thiết kế phải định hướng, hình dung cơ bản về hệ thống cần xây dựng.
- *Quá trình xây dựng phần cứng*: là một quá trình quan trọng trong việc xây dựng hệ thống. Quá trình này bao gồm việc lựa chọn các lõi xử lý cho hệ thống cũng như xây dựng các thuật toán giải quyết. Xây dựng hệ thống phần cứng được tương tác với người thiết kế thông qua các ngôn ngữ máy, ví dụ như ngôn ngữ Verilog hay VHDL. Việc xây dựng các lõi phải đảm bảo các yêu cầu đáp ứng cho mục đích ban đầu xây dựng hệ thống, việc tiêu tốn các tài nguyên xây dựng hệ thống không quá lớn và đảm bảo có thể kiểm tra hệ thống sau khi thiết kế bằng cách nạp thiết kế trên chip FPGA.
- *Quá trình xây dựng phần mềm*: quá trình xây dựng phần mềm được dựa theo các hệ thống phát triển trên phần cứng. Quá trình phát triển phần

mềm được tương tác với người dùng thông qua các ngôn ngữ lập trình bậc cao như C/C++. Tương tự như phần cứng, phần mềm được kiểm tra dựa trên các phần cứng được xây dựng trước đó.

- *Quá tình kiểm tra hệ thống*: hệ thống sau khi hoàn thiện được kiểm tra về các mặt như tổng tài nguyên hệ thống, tiêu tốn năng lượng và đáp ứng hệ thống với người dùng. Nếu không thể đáp ứng với các tiêu chí người thiết kế phải quay lại các bước thiết kế ban đầu. Nếu hệ thống đã đáp ứng đầy đủ các tiêu chí, đáp ứng các yêu cầu đặt ra cho hệ thống thì có thể đi đến bước chế tạo chip.

2.1.2 Phần cứng kit FPGA Altera DE2-115

Kit FPGA Altera DE2-115 là một trong các dòng kit DE của Altera phát triển nhằm hỗ trợ phát triển giáo dục, hiện nay kit được phổ biến hầu hết ở các trường đại học trên thế giới. Hiện nay giá hỗ trợ giáo dục của kit DE2-115 là 309$.



Hình 2. 2: Kit FPGA Altera DE2-115

Các phần cứng trên kit FPGA Altera DE2-115 [5]:

- Chip FPGA Altera Cylone IV EP4CE115F29 công nghệ 60-nm với tài nguyên 114,480 LEs.
- Chip cấu hình thiết bị EPCS64
- 1 chip SRAM 2MByte
- 2 chip DRAM, mỗi chip 32MByte
- 1 chip nhớ Flash 8Mbyte.
- Khe cắm SD Card hỗ trợ chế độ giao tiếp SPI và SD 4-bit.
- 1 cổngVGA dùng chip DAC(8 bit, 3 kênh DAC tốc độ cao), Audio CODEC (24 bit).
- TV Decoder (NTSC/PAL/SECAM) và cổng TV-in
- 1x40-pin Expansion Headers, 1 cho card HSMC.
- Clock 50Mhz, 27Mhz, 24Mhz.
- Hỗ trợ các chuẩn giao tiếp: cổng PS2, cổng USB Host/Slaver, cổng RS-232 9 chân, cổng Ethernet RJ45 hỗ trợ tốc độ tới 2Gbit, ngoài ra còn các cổng Audio.
- Các ngoại vi như: 18 switch, 4 nút nhấn, 18 led đơn đỏ, 9 led đơn xanh, 8 led 7 đoạn.
- Một module LCD 16x2.
- Có 1 bộ nhận IR.
- 2 cổng SMA dành cho clock ngõ vào hoặc ngõ ra cho người dùng.
- Nguồn cung cấp cho Kit 12V DC.

2.2 NGÔN NGỮ MÔ TẢ PHẦN CỨNG VERILOG VÀ HỆ THỐNG SoC

HDL được dùng để mô tả mô hình hóa các hệ thống số. Ngôn ngữ verilog hay VHDL là một trong hai loại ngôn ngữ được dùng phổ biến hiện nay trong các hệ thống này, đặc biệt là trong các hệ thống SoC. Trong mục người thực hiện sẽ trình bày về tổng quan về ngôn ngữ verilog và hệ thống System on Chip.

2.2.1 Ngôn ngữ mô tả phần cứng Verilog

Hình 2. 3: Minh họa ứng dụng ngôn ngữ verilog

Verilog là ngôn ngữ mô tả phần cứng [6] được sử dụng rộng rãi trong thiết kế các hệ thống số, mạch tích hợp hiện nay, cùng với ngôn ngữ VHDL, cả hai ngôn ngữ đều rất phổ biến. Verilog lần đầu được giới thiệu vào năm 1984 bởi công ty Gateway Design Automatic, đến năm 1995 Verilog được chuẩn hóa bởi tổ chức IEEE. Verilog được sử dụng phổ biến trong công nghiệp, còn VHDL được dùng nhiều trong các nghiên cứu hay lĩnh vực mang tính bảo mật cao như quốc phòng.

Một số ưu điểm của ngôn ngữ Verilog:

- *Nền tảng mạnh*: được chuẩn hóa năm 1995 từ tổ chức IEEE, phổ biến trong công nghiệp, ngôn ngữ có cấu trúc gần giống C nên gần gũi và dễ học, mô phỏng và tổng hợp hiệu quả.

- *Tính đa năng*: ngôn ngữ verilog được ứng dụng trong nhiều môi trường như thiết kế, kiểm tra, mô phỏng. Verilog cũng có thể dùng trên nhiều nền tảng như Window, Linux. Verilog phổ biến ở mức thiết kế ASIC, dễ dàng mô phỏng và tổng hợp.

- *Tính mở rộng*: Verilog có khả năng mở rộng cho phép các thiết bị bên ngoài truy cập xem thông tin chức năng thiết kế. Chính vì vậy tính bảo mật sẽ thấp hơn so với ngôn ngữ VHDL.

2.2.2 Hệ thống System on Chip (SoC)

Hệ thống System on Chip (SoC) là một hệ thống điện tử được xây dựng chung trên một khối bán dẫn với ý tưởng ban đầu là tích hợp tất cả các thành phần của một hệ thống máy tính lên cùng một chip đơn.

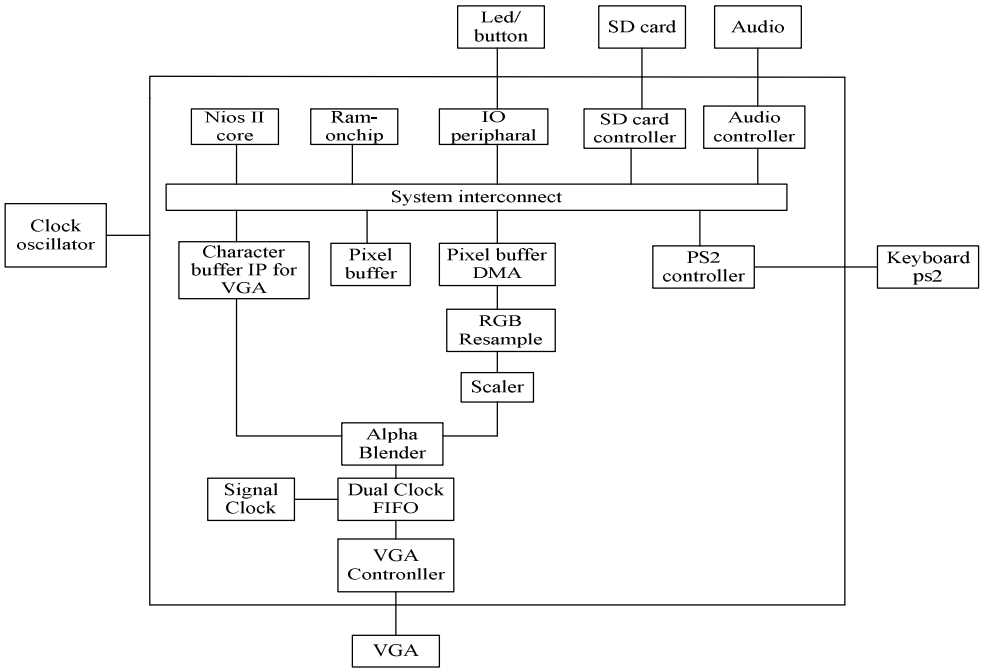
Hệ thống SoC này có thể bao gồm các khối chức năng số, tương tự, mixed-signal và các khối RF. Ứng dụng điển hình của hệ thống SoC là các hệ thống nhúng.

Một hệ thống SoC bao gồm một loạt các mạch tích hợp cho phép thực hiện các nhiệm vụ khác nhau. Các mạch tích hợp này thường là:

- Bộ vi xử lý (microprocessor)
- Bộ nhớ (RAM, ROM)
- Khối truyền thông nối thiếp UART
- Các cổng song song (parallel port)
- Khối điều khiển truy xuất bộ nhớ trực tiếp (DMA controller) …

Công nghệ thiết kế và xây dựng các hệ thống SoC có thể kể đến là:

- Công nghệ chế tạo ASIC
- FPGA



Hình 2. 4: Mô tả các thành phần một SoC

Các thiết kế SoC thường tiêu tốn ít năng lượng và có giá thành thấp hơn các hệ thống đa chip nếu so sánh cùng một thiết kế. Ngoài ra, hệ thống SoC cũng có tính ổn định cao hơn. Các ứng dụng xây dựng trên cơ sở sử dụng các hệ thống SoC cũng cho giá thành thấp hơn, không gian chiếm chỗ ít hơn.

2.3 PHƯƠNG PHÁP GIẢM NĂNG LƯỢNG CLOCK GATING

Tiết kiệm năng lượng luôn là vấn đề quan trọng trong đời sống con người hiện nay. Việc tiết kiệm năng lượng sẽ mang lại lợi ích không hề nhỏ cho mỗi chúng ta. Trong phần này chúng tôi sẽ trình bày vấn đề tiết kiệm năng lượng trong các thiết kế ASIC và giải pháp giảm công suất tiêu thụ clock gating.

2.3.1 Tiết kiệm năng lượng trong thiết kế ASIC

Ngày nay vấn đề tiêu thụ năng lượng trong các chip điện tử hay các sản phẩm điện tử luôn được quan tâm hàng đầu, điển hình như các thiết bị điện thoại di động hay các thiết bị phải liên tục di chuyển vị trí. Năng lượng tiêu thụ trên các thiết bị đó được xem như một trong các tiêu chí dùng để lựa chọn sản phẩm hiện nay.

Giải pháp của các nhà sản xuất hiện nay là tăng dung lượng chứa năng lượng của các thiết bị đồng nghĩa với việc đòi hỏi công nghệ, giá thành, kích thước của các thiết bị cũng phải tăng theo. Đó không hẳn là một giải pháp vẹn toàn cho các nhà sản xuất, chính vì vậy việc tối ưu được các thiết kế hay giải pháp giảm tiêu thụ trong các thiết kế là một vấn đề quan trọng mà nhà thiết kế phải quan tâm.

Có rất nhiều phương pháp được các nhà thiết kế đưa ra đó là: Clock Gating, Multi –Vth Cells, Multi-Voltage Supply, DVS, DVFS, Power shut-off, Memory spliting. Trong đề tài này chúng tôi sẽ trình bày về giải pháp Clock Gating.

2.3.2 Giải pháp tiết kiệm năng lượng clock gating

Clock gating [7] là kỹ thuật phổ biến hiện nay trong các thiết kế giúp giảm năng lượng tiêu thụ. Clock gating giúp giảm năng lượng tiêu thụ bằng cách tắt xung clock được cấp cho các flip-flop, bằng cách tắt các xung clock này sẽ giúp cho các flip-flop không thể chuyển trạng thái. Khi các flip-flop chuyển trạng thái

sẽ tiêu thụ một phần năng lượng và khi không thể chuyển trạng thái các flip-flop sẽ không tiêu thụ năng lượng.

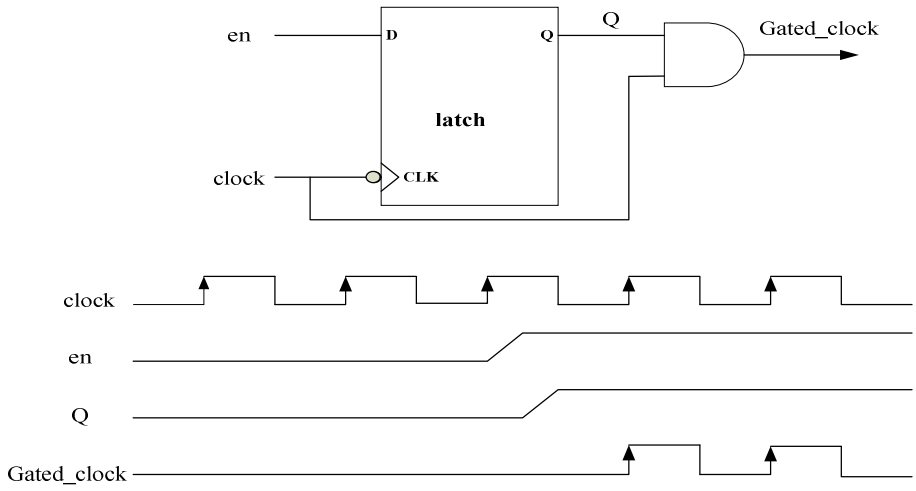
Trên lý thuyết công suất tiêu thụ của một mạch dựa trên công thức

$$P = 0.5.C.V.\propto.F \quad (1)$$

Trong đó:

- P là công suất tiêu thụ (W).
- C là điện dung tồn tại trên mạch (F).
- V là nguồn điện áp cung cấp (V).
- $\propto$ là hệ số chuyển mạch ngõ ra so với clock cung cấp (%).
- F là tần số hoạt động của clock (Hz).

Qua công thức (1) có thể thấy rằng công suất tiêu thụ của mạch phụ thuộc vào tần số hoạt động của xung clock, hệ số chuyển mạch. Như vậy việc hạn chế xung clock hoạt động khi không cần thiết sẽ giảm thiểu năng lượng tiêu hao trên hệ thống.



Hình 2. 5: Giản đồ định thời khối clock gating

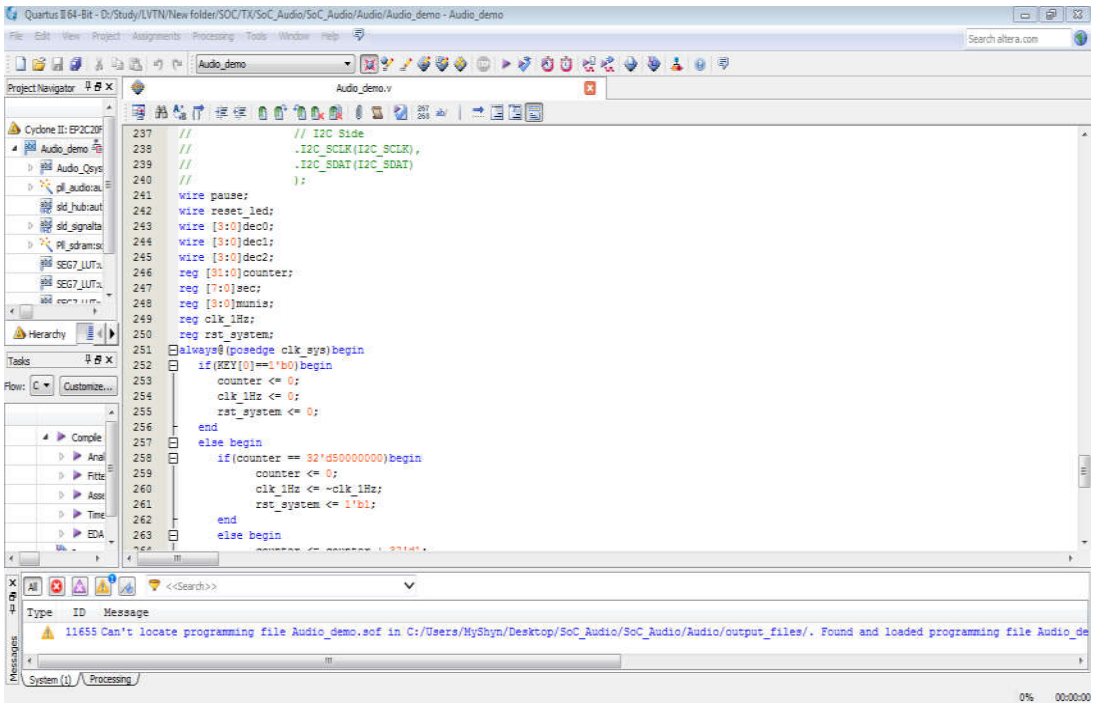
Khi tín hiệu đầu vào *en* tích cực mức cao, tín hiệu ngõ ra *Q* của flip-flop được tích cực ở mức cao. Khi đó ngõ ra của tín hiệu *gate_clock* sẽ có chu kỳ như tín hiệu *clock* ban đầu, mạch được cấp bởi tín hiệu gate_clock hoạt động bình

thường. Ngược lại khi tín hiệu *en* về mức thấp, tín hiệu ngõ ra của *gate_clock* ở mức thấp, mạch không hoạt động, không tiêu tốn năng lượng.

2.4 GIỚI THIỆU CÁC PHẦN MỀM HỖ TRỢ

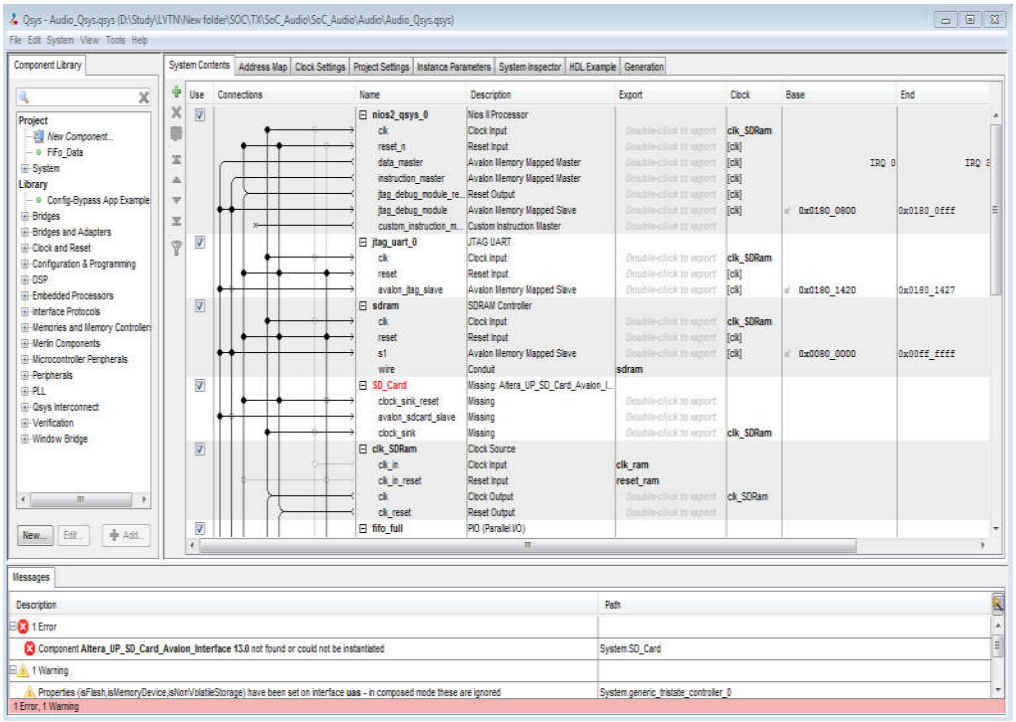
2.4.1 Quartus II

Quartus II là phần mềm được phát triển bởi hãng Altera. Đây là cộng cụ giúp người dùng có thể thiết kế và tổng hợp các mạch logic trên các linh kiện logic khả trình PLD cũng như FPGA dựa trên ngôn ngữ mô tả phần cứng của Altera, gồm các dòng APEX, Cyclone, FLEX, MAX, Stratix...



Hình 2. 6: Giao diện làm việc phần mềm Quartus II

Bên cạnh đó, phần mềm này còn hỗ trợ một số công cụ để kiểm tra thiết kế (ModemSim), xây dựng hệ thống SoC (Qsys), phân tích tín hiệu thiết kế(SignalTag)….



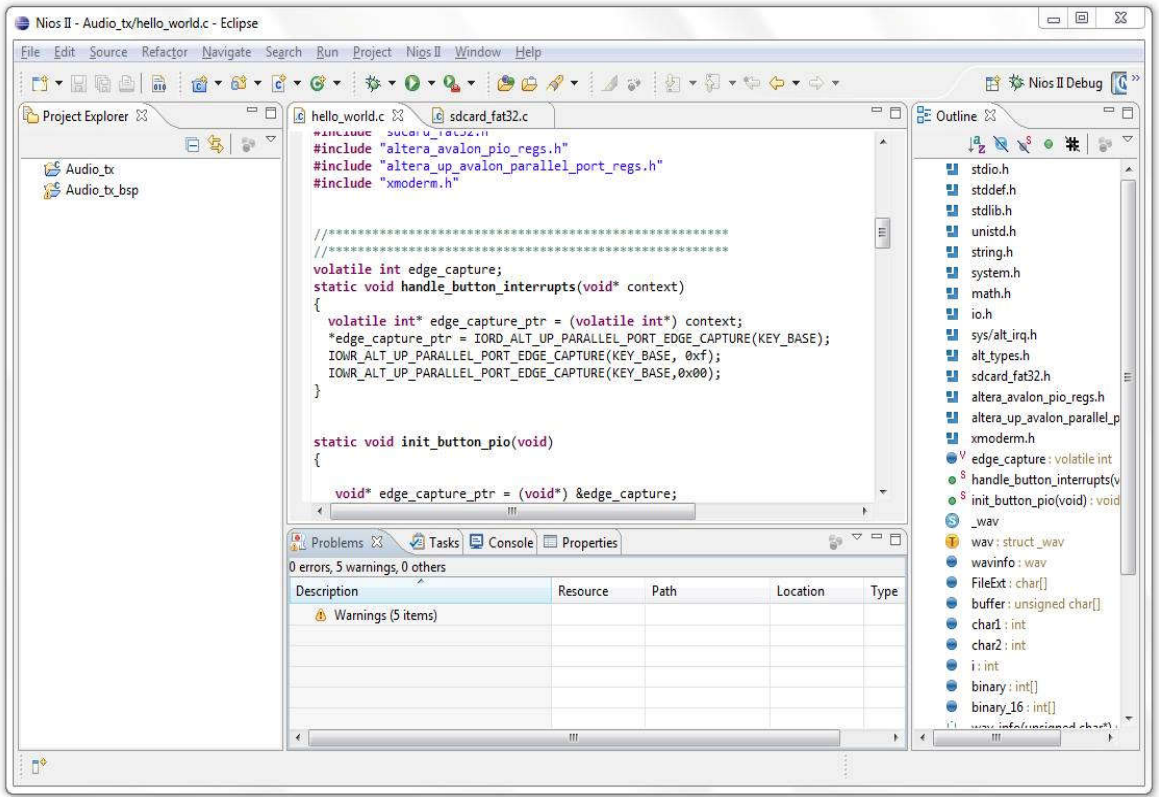
Hình 2. 7: Giao diện làm việc trên Qsys

2.4.2 Nios II Embedded Design Suite

Nios II EDS là phần mềm được phát triển và được cài đặt nhằm hỗ trợ công cụ Quartus trong việc xây dựng một hệ thống nhúng trên một SoC riêng mà người dùng xây dựng.

Ngôn ngữ lập trình được sử dụng trong Nios II EDS tương tự như C, C++ trong việc sử dụng hàm cũng như gọi các câu lệnh.

Có thể xem phần mềm này như một trình biên dịch như Keil C, điểm khác biệt nằm ở chỗ người lập trình phải hiểu rõ về hệ thống SoC mình xây dựng để gọi ra những câu lệnh, hàm cũng như sử dụng các thư viện một cách phù hợp.



Hình 2. 8: Giao diện Nios II Embedded Design Suite

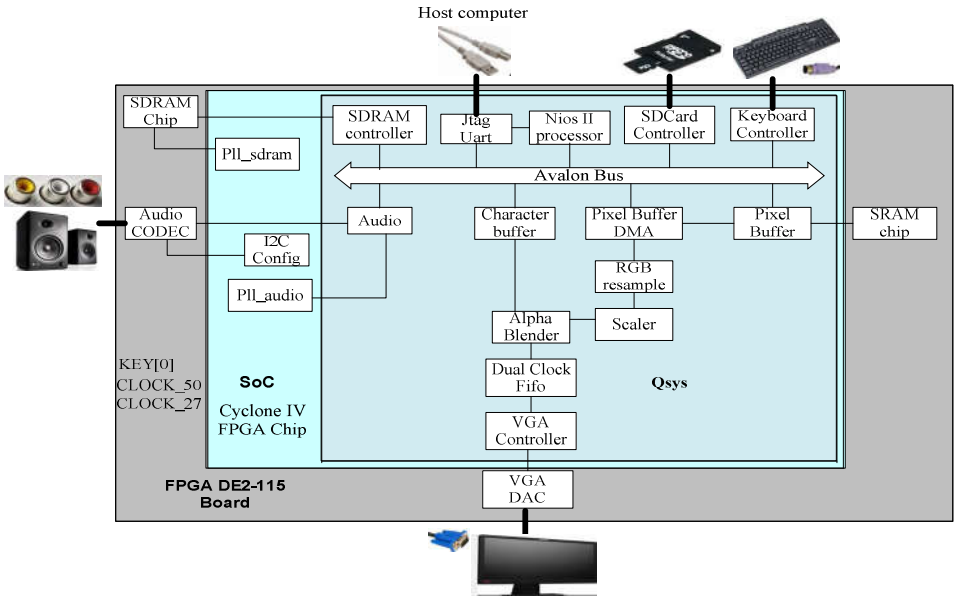
Chương 3

XÂY DỰNG HỆ THỐNG

3.1 XÂY DỰNG PHẦN CỨNG

Xây dựng phần cứng cho hệ thống dựa trên sự hỗ trợ của phần mềm chính đó là Quartus và công cụ Qsys trên Quartus. Sơ đồ xây dựng phần cứng, phương pháp thiết kế một IP dùng cho hệ thống SoC và cách thiết kế hệ thống SoC sẽ được trình bày trong phần này.

3.1.1 Sơ đồ hệ thống phần cứng



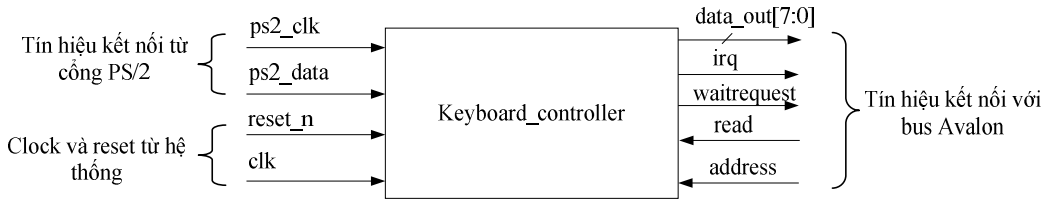
Hình 3. 1: Mô hình hệ thống phần cứng trên kit DE2-115.

Hệ thống phần cứng được xây dựng trên kit FPGA DE2-115 cùng các ngoại vi như SD Card, Loa, monitor LCD, bàn phím máy tính và máy tính. Hệ thống bao gồm cấu trúc của hệ thống SoC được xây dựng trên chip FPGA Cyclone IV với lõi xử lý trung tâm là vi xử lý Nios II và các thành phần phần cứng trên kit

FPGA như các chíp bộ nhớ SDRAM, SSRAM, các chip hỗ trợ Audio CODEC, VGA DAC. Các thành phần trên hệ thống hoạt động bằng clock 50Mhz, 27Mhz và một nút nhấn dùng để reset hệ thống nằm trên kit DE2-115.

Khi hệ thống đi vào hoạt động, vi xử lý Nios II sẽ điều khiển hệ thống kết nối, giao tiếp với các thành phần ngoại vi bao gồm VGA, Audio, Bàn phím và SD Card. Tùy thuộc vào chương trình phần mềm được giao tiếp với người dùng mà hệ thống phần cứng sẽ có các đáp ứng tương ứng.

3.1.2 Thiết kế module giao tiếp với bàn phím chuẩn PS/2

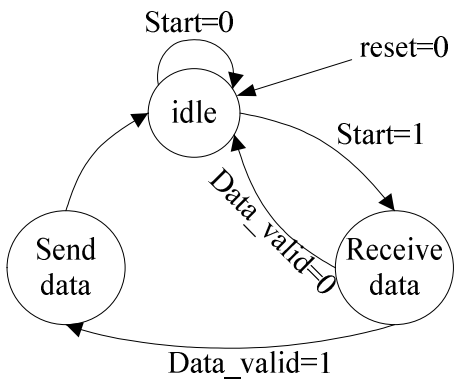


Hình 3. 2: Sơ đồ khối module keyboard_controller

Module keyboard_controller [8] được thiết kế dùng để tương tác giữa người dùng và hệ thống thông qua bàn phím máy tính chuẩn PS/2. Module có nhiệm vụ nhận dữ liệu nối tiếp từ bàn phím, sau đó chuyển dữ liệu tới hệ thống.

Bảng 1: Bảng mô tả chức năng các tín hiệu module keyboard_controller

Tín hiệu	In/Out	Chức năng các tín hiệu
clk	Input	Xung clock 50Mhz từ hệ thống cấp cho module hoạt động.
reset_n	Input	Tín hiệu reset mức thấp từ hệ thống.
ps2_data	Input	Đường dữ liệu nối tiếp từ bàn phím tới module, dữ liệu truyền đồng bộ theo xung clock ps2_clk.
ps2_clk	Input	Clock dùng để đồng bộ dữ liệu được truyền từ bàn phím.
data_out[7:0]	Output	Ngõ ra dữ liệu song song 8-bit.
irq	Output	Tín hiệu báo dữ liệu hợp lệ.
waitrequest	Output	Tín hiệu yêu cầu đợi.
read	Input	Tín hiệu đọc dữ liệu.
address	Input	Tín hiệu địa chỉ dành cho cpu quản lí các module.

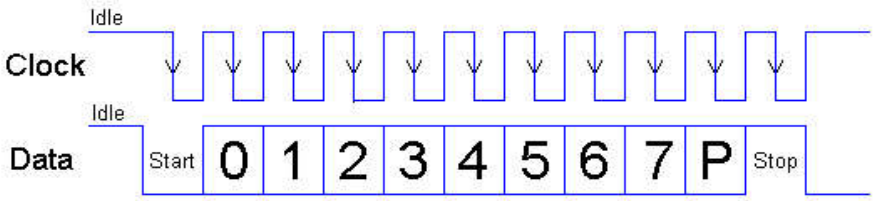


Hình 3. 3: Mô tả hoạt động module keyboard_controller bằng trạng thái máy

Ở trạng thái *idle*, module sẽ ở trạng thái chờ nhận dữ liệu khi không có dữ liệu được gửi từ bàn phím hoặc khi có tín hiệu reset module sẽ về trạng thái chờ. Khi có tín hiệu báo bắt đầu nhận dữ liệu module sẽ chuyển sang trạng thái *receive data* bắt đầu quá trình nhận dữ liệu.

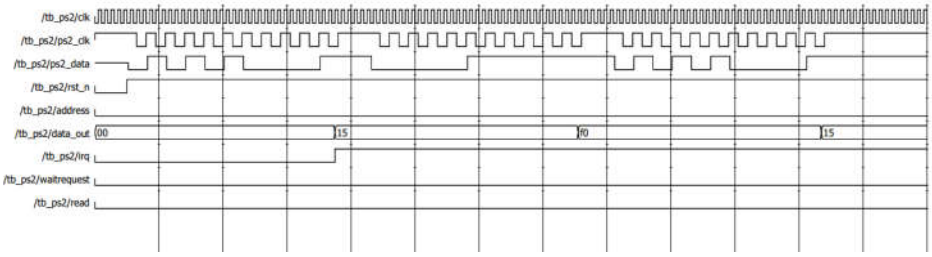
Ở trạng thái *receive data*, module sẽ bắt đầu quá trình tiếp nhận dữ liệu nối tiếp từ bàn phím. Kết thúc quá trình nhận sẽ bắt đầu kiểm tra dữ liệu, nếu dữ liệu nhận được là hợp lệ thì module sẽ chuyển sang trạng thái *send data* gửi dữ liệu về hệ thống. Sau khi truyền dữ liệu tới hệ thống module sẽ chuyển về trạng thái *idle* chờ nhận dữ liệu tiếp theo. Nếu dữ liệu nhận được bị lỗi module sẽ quay về lại trạng thái *idle.*

Giao thức truyền dữ liệu [9] từ bàn phím bao gồm 11-bit dữ liệu được truyền nối tiếp. 11-bit dữ liệu được truyền theo tứ tự bao gồm 1-bit-start(luôn ở mức thấp), 8-bit dữ liệu được truyền tiếp theo(bit có trọng số thấp được truyền trước), 1-bit kiểm tra chẵn-lẻ(bit lẻ) và cuối cùng là 1-bit-stop(luôn ở mức cao). Quá trình truyền dữ liệu được trình bày bằng giản đồ định thời trong hình 3.3.



Hình 3. 4: Giản đồ định thời khung liệu truyền từ bàn phím

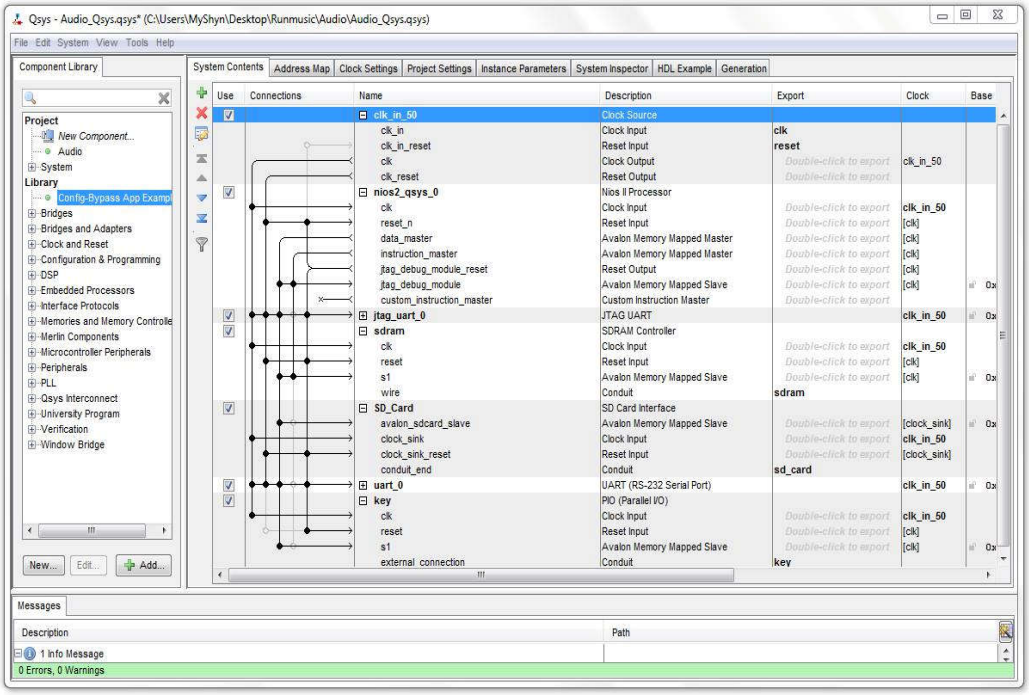
Module sau khi thiết kế có thể kiểm tra bằng cách mô phỏng bằng phần mềm ModelSim. Nếu mô phỏng đúng với quá trình truyền nhận dữ liệu, module có thể đưa vào hoạt động cùng với hệ thống SoC. Ngược lại nếu mô phỏng còn sai, người thiết kế phải thiết kế lại module.



Hình 3. 5: Dạng sóng các tín hiệu mô phỏng bằng ModelSim

3.1.3 Thiết kế hệ thống SoC

SoC được xây dựng dưa trên chip FPGA Cyclone IV với tài nguyên là 114,480 LEs. Do đó hệ thống SoC sau khi xây dựng tiêu số tốn tài nguyên phải nhỏ hơn tổng số tài nguyên có trên chip.



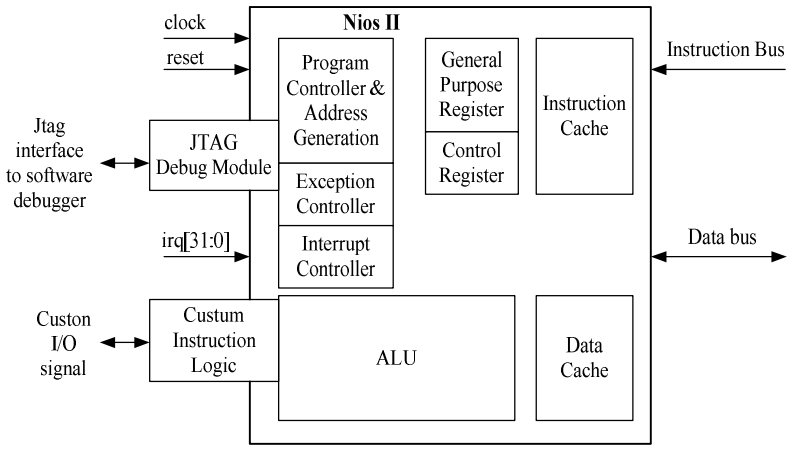
Hình 3. 6: Hệ thống SoC được xây dựng trên công cụ Qsys

Sau khi thiết kế xong các thành phần cho một hệ thống SoC, người thiết kế sẽ dùng công cụ Qsys để kết nối các thành phần hệ thống với nhau. Hệ thống sau khi thiết kế sẽ được tổng hợp và nạp xuống chip FPGA Cyclone IV thông qua cổng USB Blaster bằng phần mềm Quartus.

Các thành phần cấu tạo nên hệ thống SoC bao gồm:

❖ *Vi xử lý Nios II.*

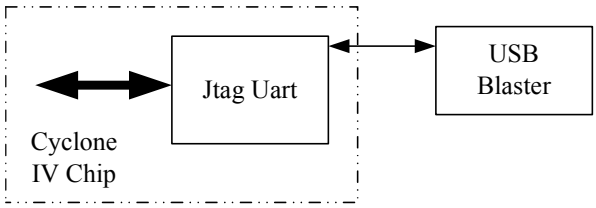
Nios II là bộ xử lý trung tâm đóng vai trò quan trọng trong một hệ thống SoC. Bên trong vi xử lý bao gồm các thanh ghi điều khiển quá trình ngắt, timer hay các khối tính toán ALU đóng vai trò quan trọng trong việc xử lý dữ liệu cho toàn hệ thống.



Hình 3. 7: Sơ đồ khối vi xử lý Nios II

❖ *Jtag Uart.*

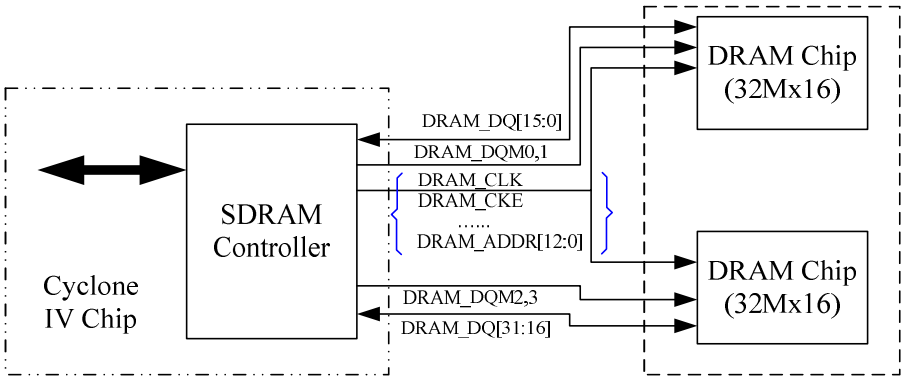
IP Jtag Uart: IP được kết nối với Nios II và hệ thống bus qua đó người dùng hệ thống có thể giám sát quá trình hoạt động cũng như debug hệ thống trong quá trình kiểm tra và vận hành hệ thống trên kit thông qua cổng USB blaster.



Hình 3. 8: Sơ đồ kết nối IP Jtag Uart

❖ *IP điều khiển SDRAM.*

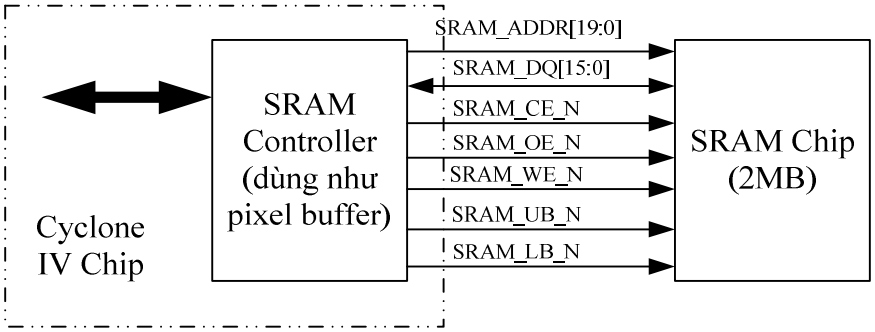
IP SDRAM Controller: IP có chức năng quan trọng trong việc vận hành hệ thống và giao tiếp với Nios II. IP có nhiệm vụ lưu trữ dữ liệu từ hệ thống tới bộ nhớ trên chip SDRAM hoặc đưa dữ liệu từ SDRAM chip tới hệ thống. IP này được kết nối giao tiếp với hai chip SDRAM có tổng dung lượng là 128Mbyte.



Hình 3. 9: Sơ đồ kết nối IP SDRAM controller và chip SDRAM

❖ *IP điều khiển SRAM.*

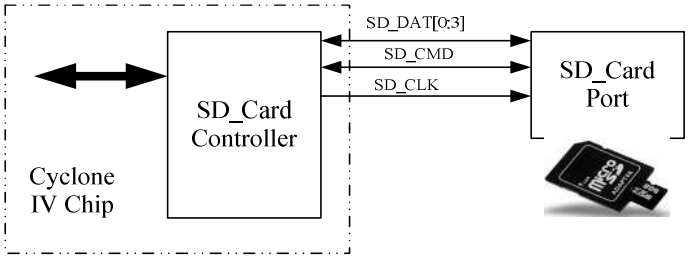
IP Pixel buffer: IP pixel_buffer có chức năng lưu trữ một khung ảnh vào SRAM chip và đưa dữ liệu ra khi cần. Chip SRAM có dụng lượng là 2Mbyte.



Hình 3. 10: Sơ đồ kết nối giữa IP pixel buffer và chip SRAM

❖ *IP điều khiển SD Card.*

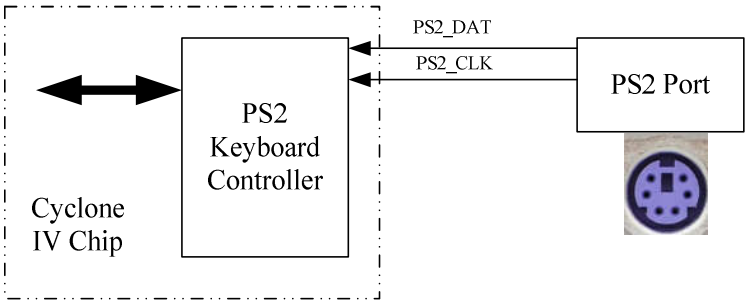
IP SD Card Controller [10] có chức năng truyền/nhận dữ liệu từ thiết bị SD Card và giao tiếp với Nios II thông qua bus Avalon. IP hỗ trợ giao tiếp với SD Card có dụng lượng tối đa là 2GB.



Hình 3. 11: Sơ đồ kết nối giữa IP SD Card Controller và SD Card

❖ *IP giao tiếp với bàn phím*

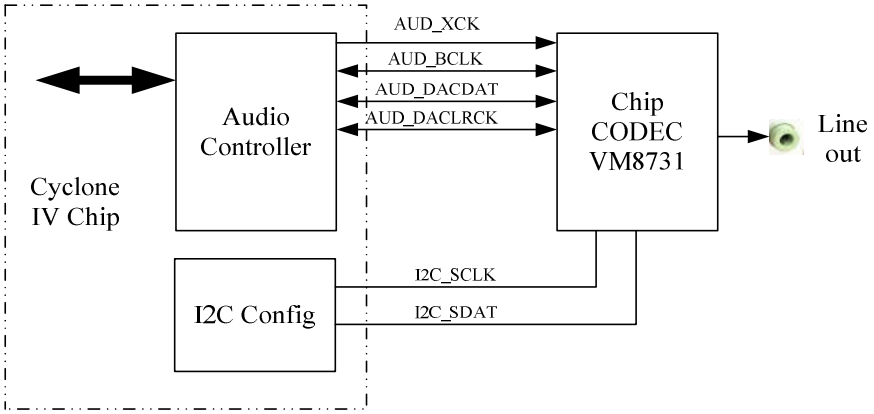
IP Keyboard_controller được thiết kế để kết nối với bàn phím máy tính chuẩn PS/2 thông qua PS/2 Port và thực hiện truyền dữ liệu tới bus thông qua sự giám sát của Nios II.



Hình 3. 12: Sơ đồ kết nối giữa IP PS/2 và bàn phímPS/2

❖ *IP điều khiển Audio*

IP Audio [11] được điều khiển bởi Nios II có chức năng đưa dữ liệu tới chip Audio CODEC và đưa tới loa. IP hoạt động dựa trên 2 xung clock là 50Mhz từ hệ thống và 18.432 Mhz từ PLL dùng cho việc phát các file nhạc ra loa. Để chip Audio CODEC hoạt động được phải cần thêm một module cấu hình chức năng là I2C_Config, module cấu hình chip thông qua chuẩn giao tiếp I2C.



Hình 3. 13: Sơ đồ cấu hình cho hệ thống Audio

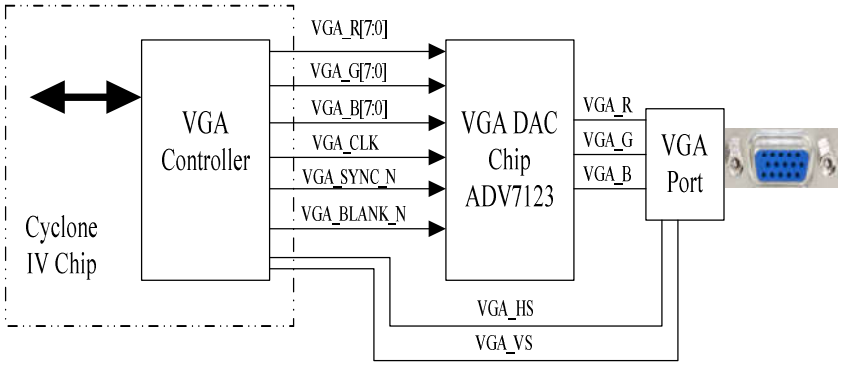
❖ *IP điều khiển VGA*

IP VGA_Controller có chức năng đưa dữ liệu từ IP Dual_Clock_FIFO tới chip xử lý dành riêng cho VGA và đưa tới màn hình LCD thông qua cổng VGA. Clock cung cấp chuẩn cho VGA thường có tần số 25.175MHz, clock được tính dựa trên công thức:

$$F = [640(\text{pixel/row}) + 160] \times [480\ (\text{pixel/col}) + 45] \times 60(\text{refreshes/s}) \quad (2)$$

Trong đó:

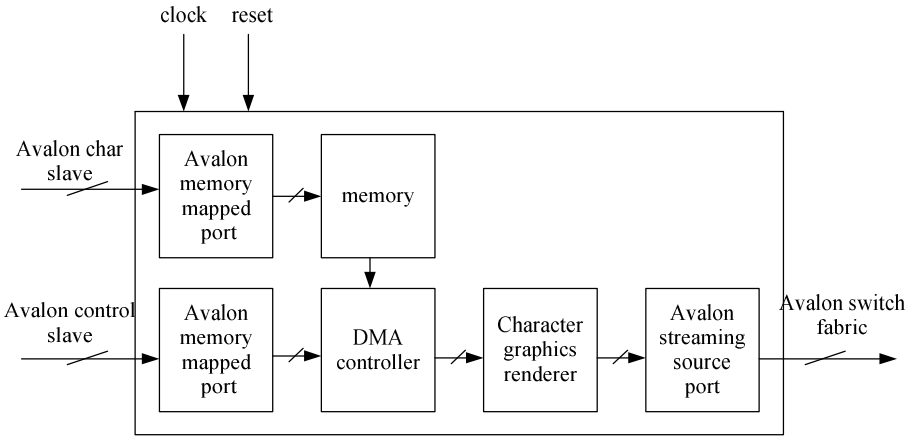
- 640 là số pixel theo hàng, 480 là số pixel theo cột
- 160, 45 là thời gian bắt đầu và kết thúc quét 1 hang và cột (s).
- 60 là tốc độ làm tươi tiêu chuẩn cho một màn hình (Hz, 60 khung hình/s).



Hình 3. 14: Sơ đồ kết nối giữa VGA và hệ thống SoC

❖ *IP xử lý ký tự.*

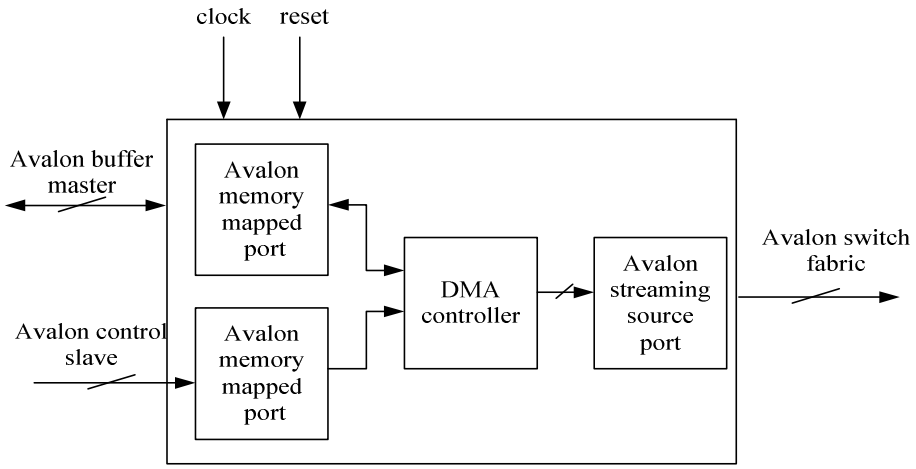
IP character buffer: IP đưa kí tự tới màn hình VGA. IP được điều khiển bởi Nios II thông qua bus. Khi Nios II thực hiện gửi ký tự ASCII đến IP character buffer, IP lưu ký tự đó lên bộ nhớ, sau đó khối DMA Controller sẽ đọc ký tự từ bộ nhớ và gửi đến khối character renderer. Khối character renderer sẽ chuyển ký tự mã ASCII sang định dạng pixel và đưa tới ngõ ra của IP. Ngõ ra của IP được kết nối với IP Alpha Blender.



Hình 3. 15: Sơ đồ khối IP character buffer

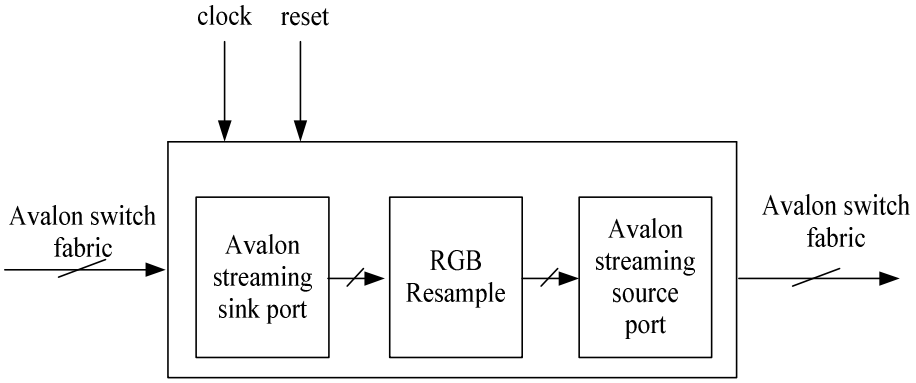
❖ *Các IP xử lý pixel.*

IP Pixel_buffer_dma: IP được điều khiển bởi vi xử lý Nios II, IP có chức năng đọc dữ liệu được chứa trong bộ nhớ pixel_buffer và đưa tới IP RGB_Resample tiếp tục xử lý.



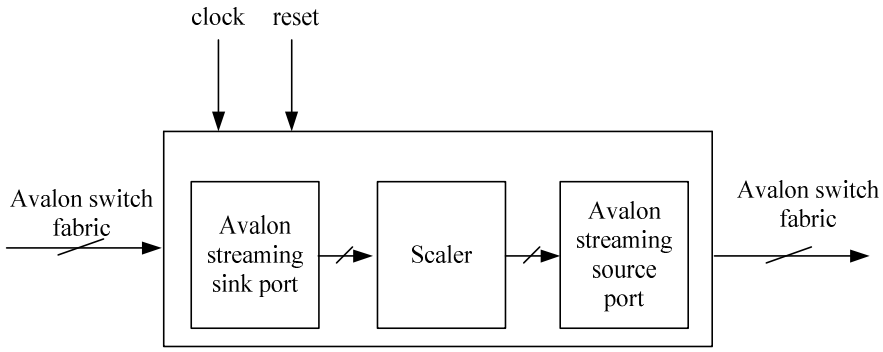
Hình 3. 16: Sơ đồ khối IP pixel buffer dma

IP RGB_Resample: IP có chức năng chuyển đổi định dạng không gian màu từ 16-bit RGB ngõ vào thành 30-bit RGB ngõ ra. IP sẽ chuyển dữ liệu nhận được từ IP Pixel_buffer_dma tới IP Scaler dựa trên clock hoạt động là 50Mhz.



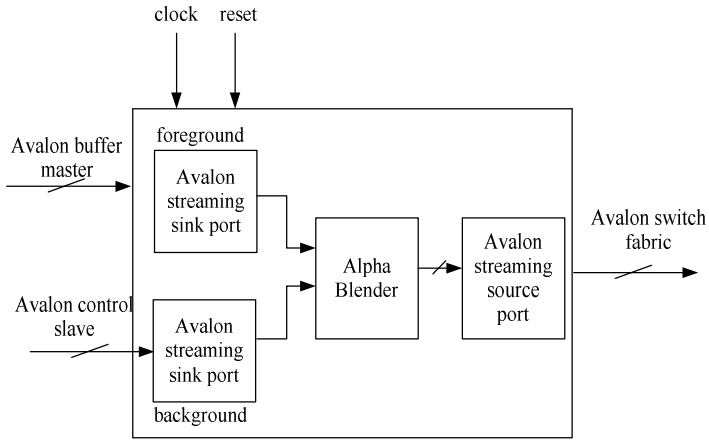
Hình 3. 17: Sơ đồ khối IP RGB resample

IP Scaler: IP có chức năng chuyển độ phân giải của một khung ảnh có kích thước từ 320x240 sang 640x480. IP được kết nối với ngõ vào từ IP RGB_Resample vã ngõ ra đưa tới IP Alpha_Blender, hoạt động với clock là 50Mhz.



Hình 3. 18: Sơ đồ khối IP Scaler

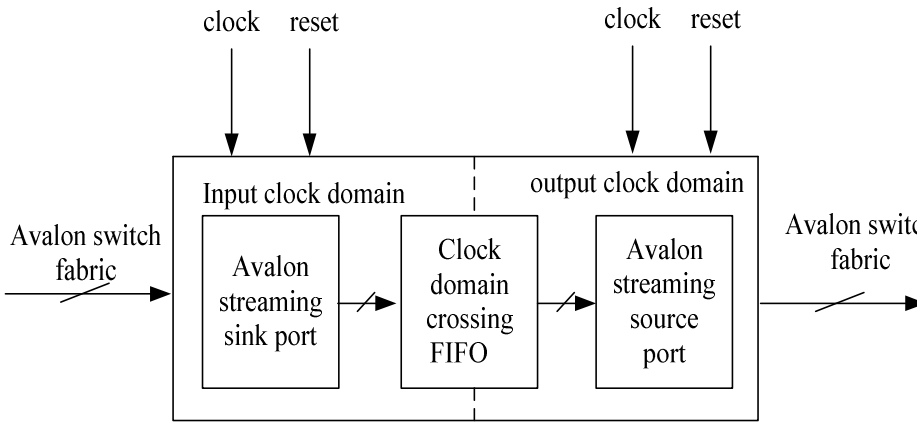
IP Alpha Blender: IP có chức năng tổng hợp 2 tín hiệu ngõ vào thành một tín hiệu ngõ ra. Tín hiệu ngõ vào đước sử dụng trong hệ thống là tín hiệu truyền ký tự có độ rộng 40bit và tín hiệu truyền khung ảnh có độ rộng 30bit, hai tín hiệu này sẽ được tổng hợp thành một tín hiệu ngõ ra có độ rộng 30bit. IP sử dụng clock 50Mhz từ hệ thống. Sau khi được tổng hợp thành một tín hiệu dữ liệu sẽ được chuyển tới IP Dual_Clock_FIFO



Hình 3. 19: Sơ đồ khối IP Alpha Blender

❖ IP *Dual Clock FIFO*.

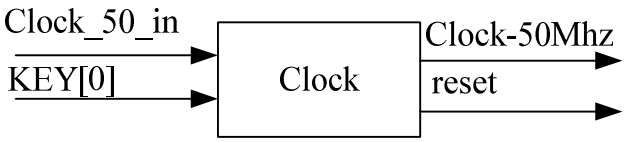
IP có chức năng nhận dữ liệu được chuyển tới từ IP Alpha Blender và chuyển tiếp tới IP VGA Controller để đưa ra LCD. IP này dùng để chuyển tiếp dữ liệu giữa từ miền xung clock 50Mhz sang miền xung clock 25.175Mhz dùng trong VGA.



Hình 3. 20: Sơ đồ khối IP Dual Clock FIFO

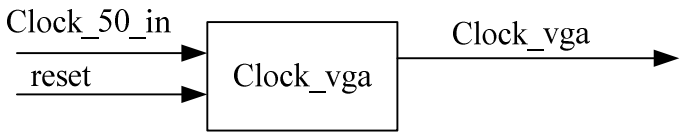
❖ *Các bộ chia xung clock*

Bộ ổn định xung clock 50Mhz dùng cho toàn hệ thống, module được kết nối với xung clock ngõ vào có tần số 50Mhz và nút nhấn KEY[0] trên kit. Khi nhấn nút KEY[0] trên kit sẽ reset toàn bộ hệ thống SoC.



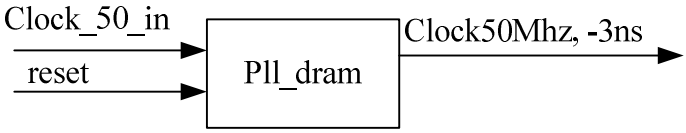
Hình 3. 21: Bộ ổn định xung clock 50Mhz cho hệ thống

Bộ chia xung clock_vga dùng cho 2 IP là IP Dual clock fifo và IP VGA Controller, do 2 IP hoạt động với clock chuẩn dành cho vga là 25.175Mhz. Clock cho vga được chia từ clock 50Mhz lấy từ bộ ổn định clock 50Mhz của hệ thống.



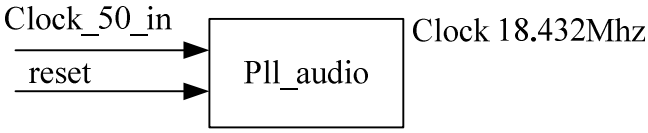
Hình 3. 22: Bộ chia clock dùng cho vga

Bộ chia xung clock pll_dram dùng cho SDRAM. SDRAM sử dụng clock 50Mhz cùng với hệ thống nhưng có độ trễ hơn -3ns.



Hình 3. 23: Bộ chia xung clock cho chip SDRAM

Bộ chia xung clock pll_audio dùng cho IP Audio. IP sử dụng clock 50Mhz chia thành xung clock có tần số18.432Mhz để phục vụ đưa dữ liệu ra cổng audio.

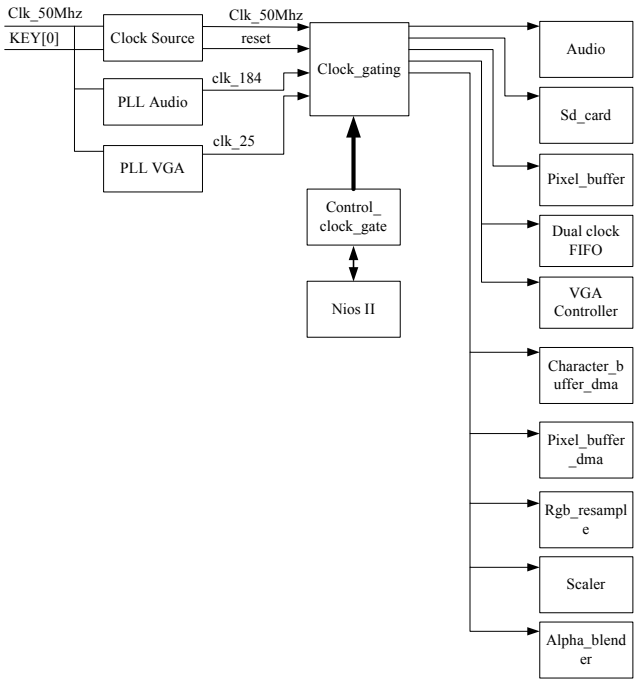


Hình 3. 24: Bộ chia xung clock cho hệ thống Audio

3.1.4 Thiết kế hệ thống clock gating

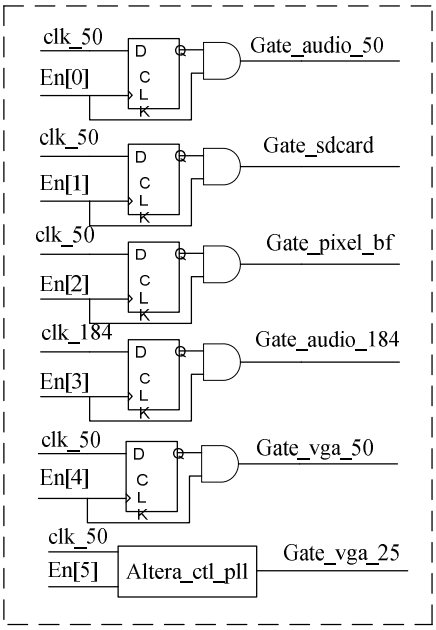
Hệ thống clock gating được thiết kế dựa trên nền tảng hệ thống SoC đã thiết kế trước đó. Hệ thống này được thiết kế để đưa các khối phần cứng gồm: IP character buffer, IP pixel buffer, IP SD Card Controller, IP Audio, IP dual clock fifo, VGA Controller và một số IP khác về chế độ nghỉ khi hệ thống không hoạt động. Hệ thống có thể được điều khiển thông qua chương trình phần mềm.

Hệ thống clock gating trong thiết kế chỉ bao gồm 2 khối nhỏ nên chỉ chiếm thêm một phần nhỏ số cổng LEs (khoảng 60LEs) trong thiết kế nên không ảnh hưởng nhiều đến diện tích của hệ thống sau khi thiết kế.



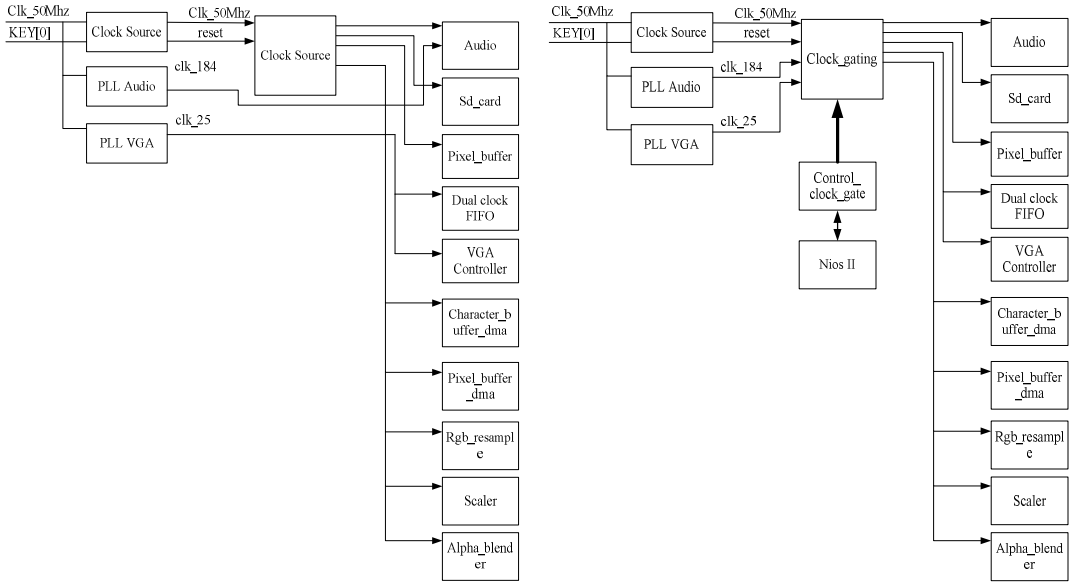
Hình 3. 25: Sơ đồ thiết kế clock gating cho hệ thống SoC

Khối clock gating sẽ hoạt động thông qua tín hiệu bật tắt từ khối điều khiển control_clock_gate, khối điều khiển sẽ hoạt động thông qua giám sát từ Nios II, Nios có thể điều khiển bật/tắt các xung clock cho từng khối.



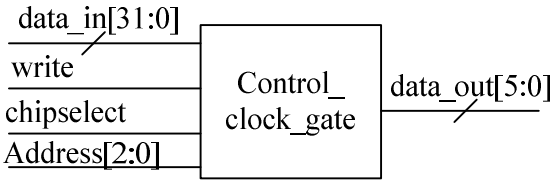
Hình 3. 26: Sơ đồ bên trong khối clock gating

Khối clock_gating sẽ bao gồm các cell clock gating dùng để điều khiển từng xung clock đến từng IP trong hệ thống. Các cell sẽ được điều khiển bằng khối control_clockgate. Khối này sẽ điều khiển các cell clockgate bằng tín hiệu En[0:5], tín hiệu En sẽ được điều khiển giám sát thông qua Nios,. Khi hệ thống khởi động, khối control_clockgate sẽ reset các tín hiệu En[0:5] này ở mức cao để khởi động hệ thống ban đầu. Khi người dùng chọn chế độ nghỉ cho hệ thống thì khối control_clockgate sẽ điều khiển các tín hiệu En về mức thấp, thông qua đó các clock cung cấp cho các IP sẽ được ngắt và giữ ở mức thấp.



(a) Hệ thống chưa có clock gate **(b)** Hệ thống đã thiết kế clock gate

Hình 3. 27: Sơ đồ hệ thống trước (a) và sau (b) khi thiết kế clock gate



Hình 3. 28: Sơ đồ chức năng khối control_clock_gate

Khối control_clock_gate được sử dụng trong thiết kế, khối có nhiệm vụ nhận dữ liệu từ vi xử lý Nios II dùng làm tín hiệu điều khiển các tín hiệu kích hoạt

hoặc vô hiệu hóa các clock gate. Khối này gồm: ngõ ra dữ liệu 6bit tương ứng với các tín hiệu kích hoạt/vô hiệu hóa clock gate, tín hiệu data_in được kết nối với Nios II thông qua bus Avalon và các tín hiệu đi kèm như tín hiệu write, hay chipselect(lựa chọn chip) hay các tín address(địa chỉ). Ngoài ra khối còn có chức năng tự động thiết lập ngõ ra data_out ở mức 1 khi mạch bắt đầu hoạt động từ trạng thái reset của hệ thống. Khối này sử dụng clock 50MHz và các tín hiệu reset của hệ thống.

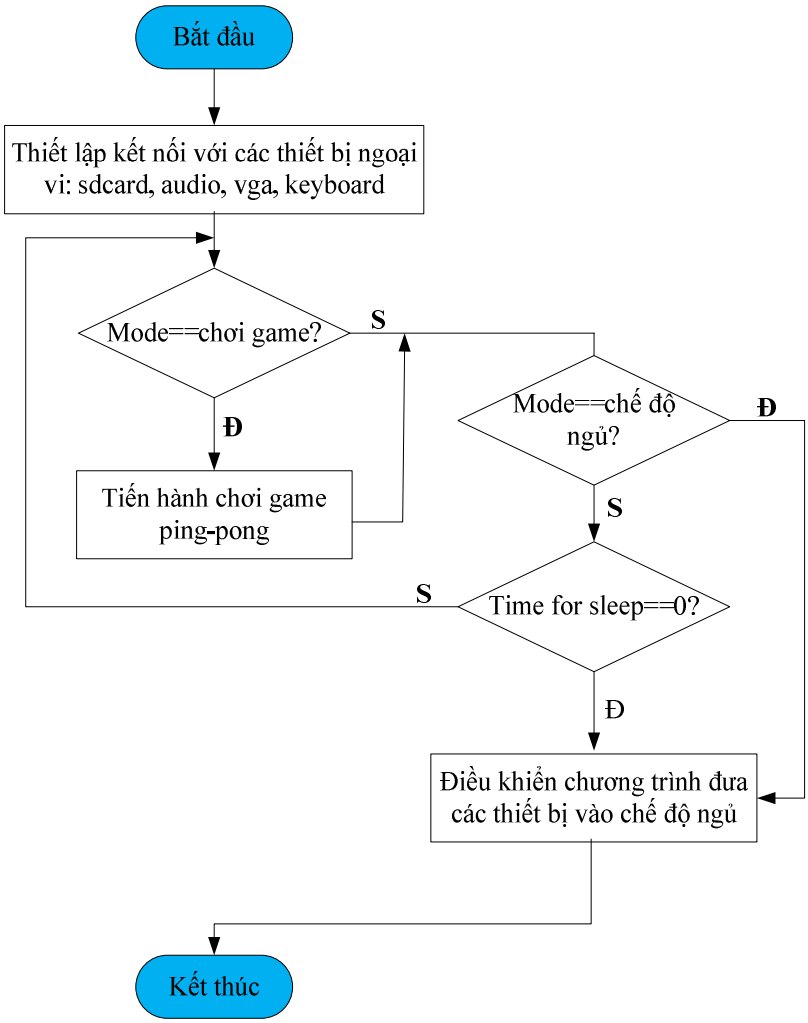
Các thiết kế bên trong khối control_clock_gate bao gồm các flip-flop nhằm giữ dữ liệu ngõ ra và một số mạch tổ hợp được thiết. Do vậy, thiết kế khối control_clock_gate không chiếm quá nhiều diện tích trong thiết kế và ảnh hưởng đến quá nhiều đến tiêu tốn năng lượng hệ thống.

3.2 XÂY DỰNG PHẦN MỀM

Phần mềm hoạt động dựa trên phần cứng SoC được thiết kế nhờ sự hỗ trợ từ công cụ EDS, công cụ sẽ hỗ trợ biên dịch phần mềm và nạp xuống phần cứng thông qua cổng USB Blaster được kết nối với máy tính cá nhân. Trong mục này sẽ trình bày các sơ đồ hệ thống tổng quan phần mềm và các lưu đồ xây dựng ứng dụng dựa trên phần cứng SoC.

3.2.1 Sơ đồ hệ thống chương trình

Hệ thống chương trình được xây dựng trên công cụ EDS, công cụ này sẽ hỗ trợ cho người dùng có thể lập trình phần mềm bằng ngôn ngữ C/C++. Bên cạnh đó công cụ còn cung cấp các các phương thức giao tiếp, driver hỗ trợ giao tiếp với phần cứng. Ngoài ra người dùng các phương thức giao tiếp hay các driver do chính người dùng thiết kế cho hệ thống.



Hình 3. 29: Sơ đồ hoạt động của phần mềm

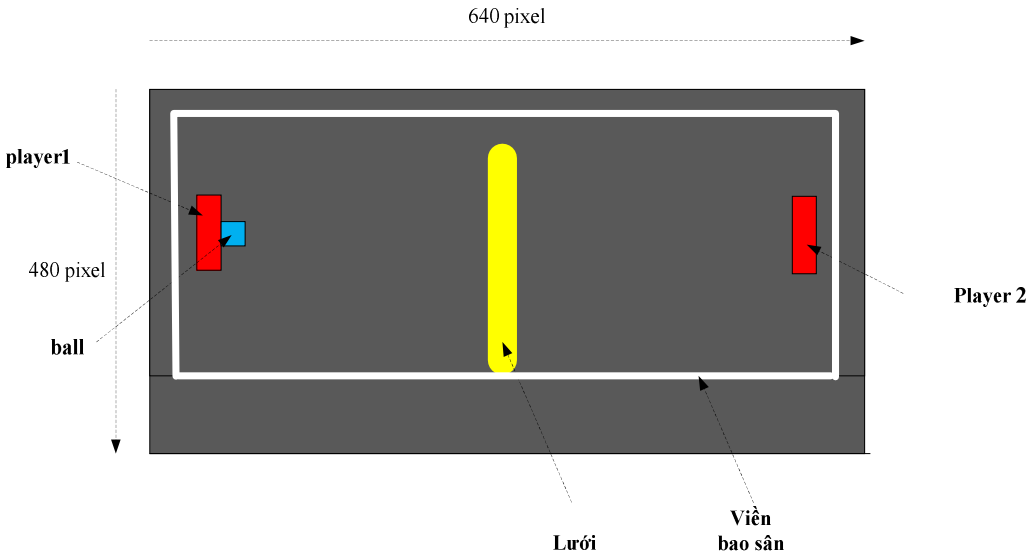
Trên hệ thống phần mềm, chúng tôi xây dựng một chương trình với ngôn ngữ C. Chương trình này sẽ điều khiển quá trình giao tiếp của Nios II với các thiết bị ngoại vi hay giữa các khối phần cứng với nhau. Người dùng có thể tương tác với hệ thống thông qua màn hình LCD dùng để hiển thị quá trình hoạt động của hệ thống hoặc bàn phím chuẩn PS/2 dùng để điều khiển hệ thống thông qua lập trình ngắt.

Sau khi chương trình bắt đầu hoạt động, chương trình sẽ bắt đầu thực hiện kết nối, quản lý hoạt động các thiết bị phần cứng như SDRAM, SRAM, điều khiển các IP hoạt động và giao tiếp với các thiết bị ngoại vi. Khi quá trình kết nối

kết thúc, người dùng có thể lựa chọn một trong hai chế độ hoạt động cho hệ thống là chơi game hoặc đưa hệ thống về chế độ ngủ đông nếu không dùng tới hệ thống. Nếu người dùng không tương tác tới hệ thống, sau một khoảng thời gian thì phần mềm sẽ điều khiển hệ thống về chế độ ngủ. Người dùng có thể khởi động lại hệ thống thông qua bàn phím hoặc reset hệ thống phần cứng.

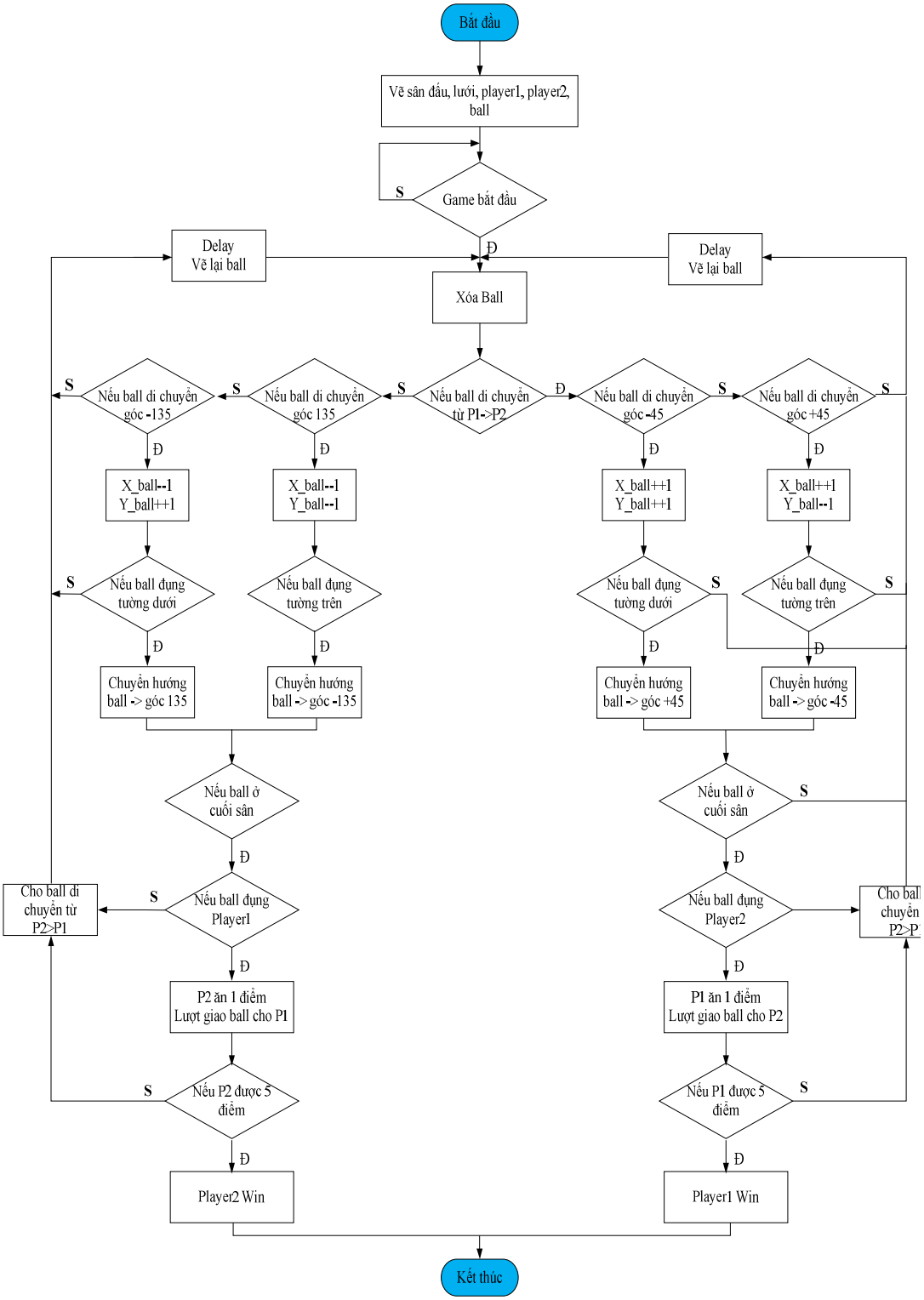
3.2.2 Ứng dụng trò chơi ping-pong

Ứng dụng trò chơi ping-pong tương tác với người dùng thông qua màn hình LCD được kết nối với hệ thống thông qua cổng VGA, người dùng hệ thống điều khiển trò chơi bằng bàn phím máy tính chuẩn PS/2 thông qua cổng kết nối PS/2.



Hình 3. 30: Minh họa trò chơi ping-pong trên màn hình LCD

Trò chơi ping-pong được thiết kế dựa theo môn thể thao bóng bàn trên thực tế, trò chơi bao gồm hai người chơi được minh họa bằng hình chữ nhật màu đỏ. Màu vàng giữa được thiết kế như lưới của môn thể thao bóng bàn, hình vuông màu xanh dương là quả bóng và toàn bộ sân đấu được bao viền bằng màu trắng. Nếu một trong hai người chơi để bóng vượt tới cuối sân của người chơi đó thì đối người đối diện sẽ đạt được một điểm trong trò chơi và quền giao bóng được giao cho người vừa mất điểm. Nếu một người đạt tới 5 điểm trước thì người chơi đó sẽ chiến thắng, trò chơi kết thúc.

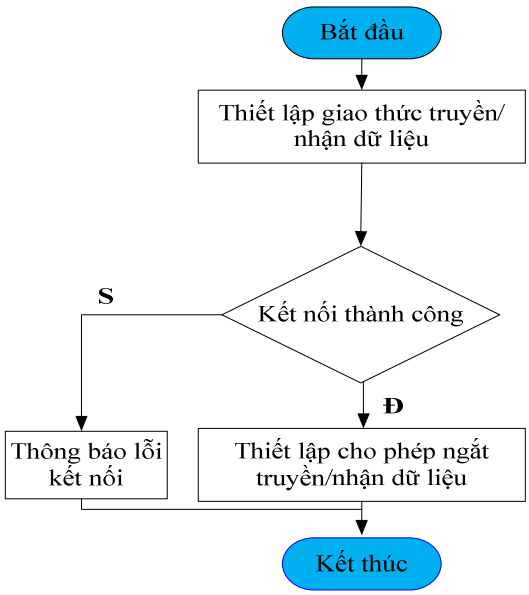


Hình 3. 31: Lưu đồ thuật toán trò chơi ping-pong

Lưu đồ trên thể hiện quá trình hoạt động của trò chơi ping-pong được xây dựng bằng ngôn ngữ C. Giải thuật xây dựng trò chơi dựa trên quá trình xóa và vẽ các điểm ảnh. Mỗi quá trình ball di chuyển tương ứng với quá trình xóa tại vị trí pixel cũ và vẽ lại tại vị trí mới với độ delay chương trình. Ball di chuyển bao gồm 4 hướng, hướng góc 45^0, hướng góc -45^0, hướng góc 135^0 và góc -135^0. Nếu ball vượt qua người chơi thì ball sẽ được vẽ lại tại vị trí người vừa thua, người đối diện sẽ ăn được một điểm. Trò chơi kết thúc khi một trong hai người chơi đạt 5 điểm.

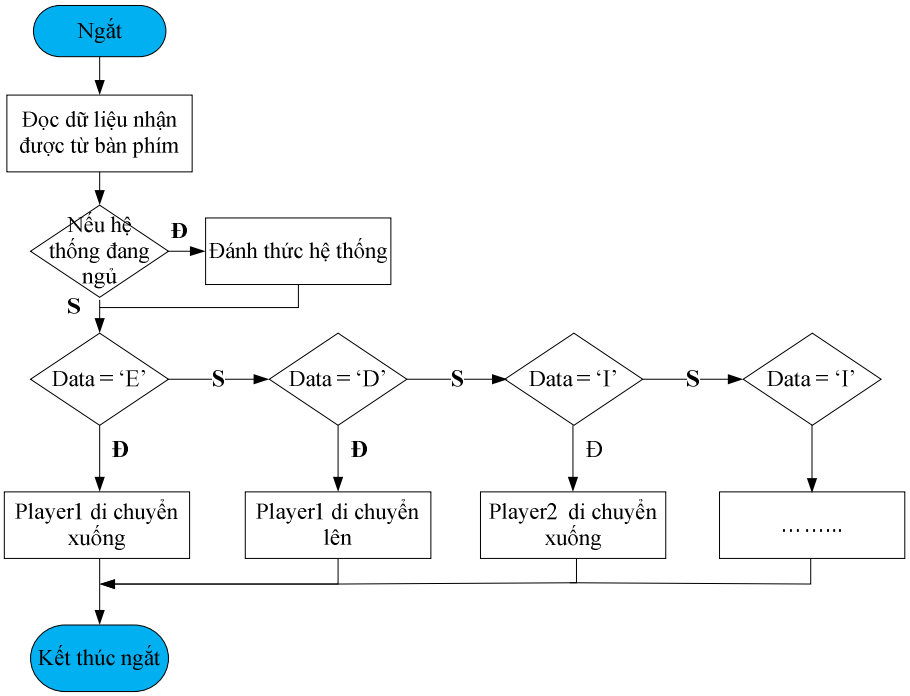
3.2.3 Sơ đồ hoạt động của các thiết bị ngoại vi

❖ *Ngoại vi bàn phím PS/2*



Hình 3. 32: Lưu đồ chương trình kết nối với bàn phím PS/2

Thiết bị ngoại vi bàn phím PS/2 được kết nối với hệ thống SoC thông qua cổng kết nối PS/2 nằm trên kit. Ngoại vi này được điều khiển bởi phần cứng keyboard_controller được thiết kế trong hệ thống SoC và kết nối tới hệ thống điều khiển ngắt của Nios II. Khi có dữ liệu từ bàn phím gửi tới hệ thống, chương trình phục vụ ngắt cho bàn phím sẽ thực thi.

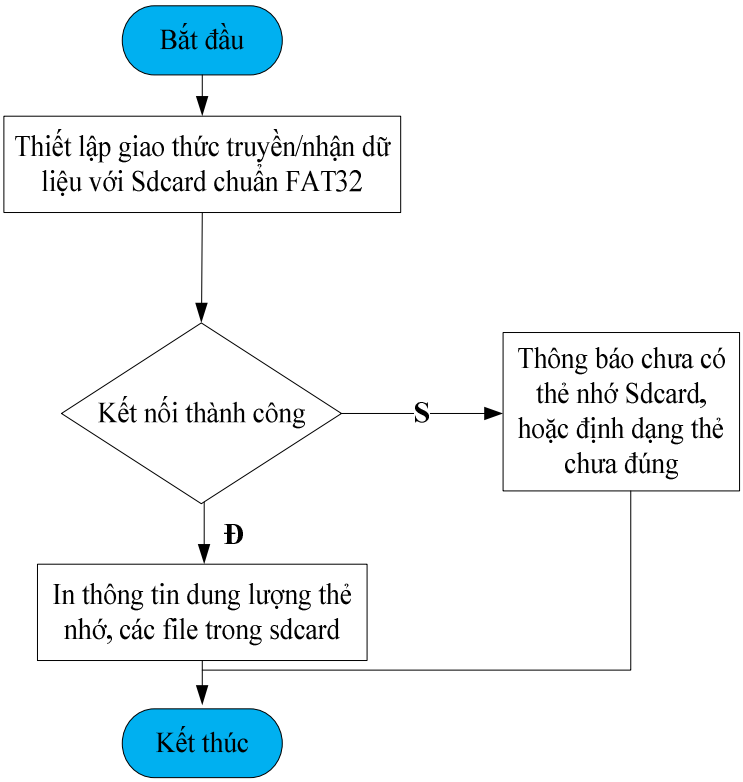


Hình 3. 33: Lưu đồ chương trình phục vụ ngắt bàn phím

Khi có dữ liệu gửi tới bàn phím, một chương trình phục vụ ngắt sẽ được thực thi. Khi bắt đầu chương trình ngắt, nếu hệ thống đang trong quá trình ngủ đông thì chương trình ngắt sẽ điều khiển hệ thống quay lại chế độ hoạt động. Ứng với dữ liệu được nhận từ bàn phím chương trình sẽ có các đáp ứng tương ứng và kết thúc quá trình ngắt.

❖ *Ngoại vi SD Card*

Thiết bị SD Card có dụng lượng là 1Gbyte định dạng FAT32 được kết nối tới hệ thống thông qua cổng SD Card và sử dụng chế độ SPI để giao tiếp với hệ thống. Ngoại vi SD chứa các dữ liệu là các file nhạc dùng cho ứng dụng trò chơi ping-pong với định dạng nhạc là .wav.

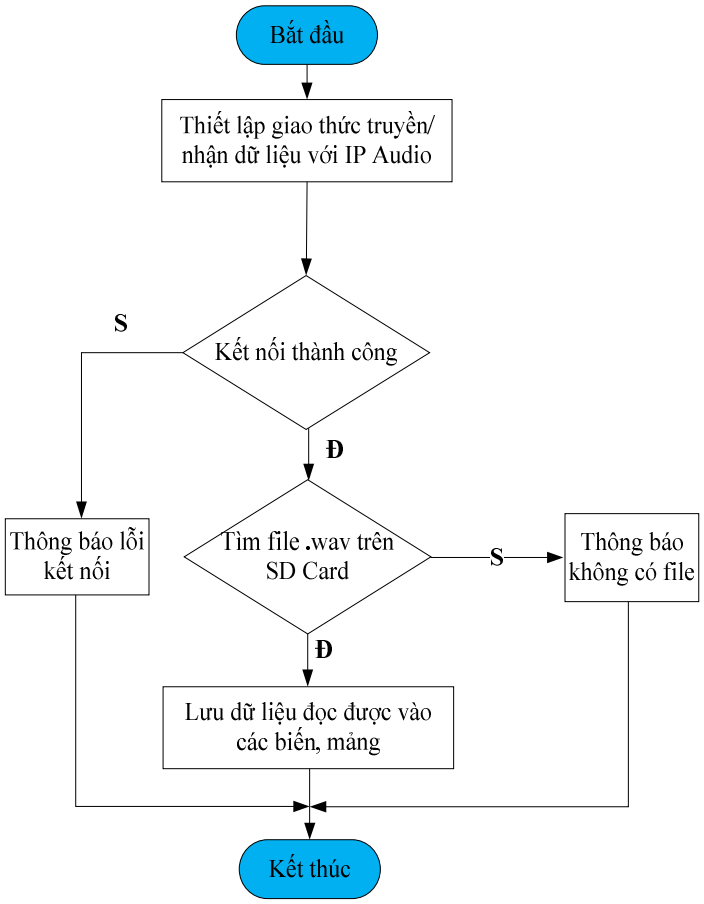


Hình 3. 34: Lưu đồ chương trình kết nối SD Card

Khi hệ thống đi vào hoạt động, một chương trình kiểm tra kết nối tới SD Card sẽ được thực thi. Nếu không có thẻ nhớ được kết nối với hệ thống hoặc thẻ nhớ không đúng định dạng là FAT32 thì chương trình sẽ thông báo. Ngược lại, chương trình sẽ đưa ra thông tin các file trên thẻ nhớ, dung lượng thẻ.

❖ *Ngoại vi Audio*

Thiết bị loa được kết nối với hệ thống thông qua cổng line-out của Audio. Ngoại vi được điều khiển bởi phần cứng là IP Audio và chip Audio CODEC. Khi chương trình đi vào hoạt động, một chương trình con sẽ kiểm tra quá trình kết nối tới hệ thống phần cứng audio sẽ được thực thi.

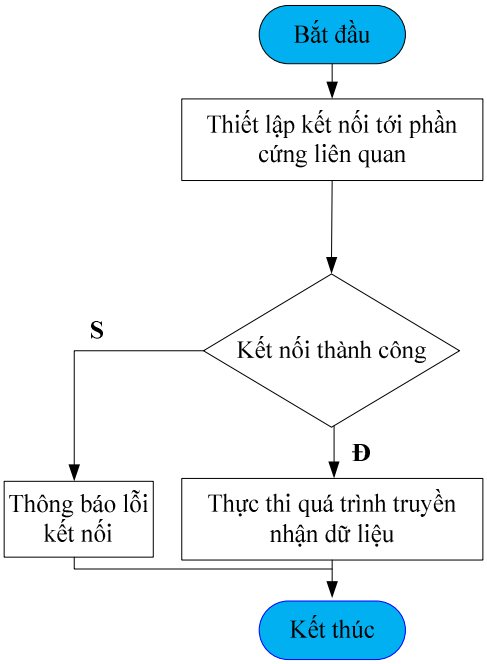


Hình 3. 35: Lưu đồ chương trình kết nối audio

Khi chương trình được thực thi, Nios II sẽ kiểm tra kết nối tới IP Audio, nếu kết nối thành công chương trình sẽ tìm các file nhạc có định dạng .wav. Nếu không tìm thấy các file, chương trình sẽ thông báo, ngược lại chương trình sẽ lưu các giá trị vừa tìm được vào các biến trên chương trình và kết thúc quá trình kết nối.

❖ *Ngoại vi VGA*

Trong đề tài này, chúng tôi sử dụng màn hình LCD 21'inch để kết nối tới hệ thống thông qua cổng VGA. Hệ thống VGA được điều khiển bởi Nios II thông qua các IP phần cứng là character buffer, pixel buffer, pixel buffer dma, rgb resample, scaler, dual clock fifo, vga controller. Khi hệ thống đi vào hoạt động, một chương trình con sẽ kiểm tra kết nối tới các phần cứng này.



Hình 3. 36: Lưu đồ chương trình kết nối các phần cứng VGA

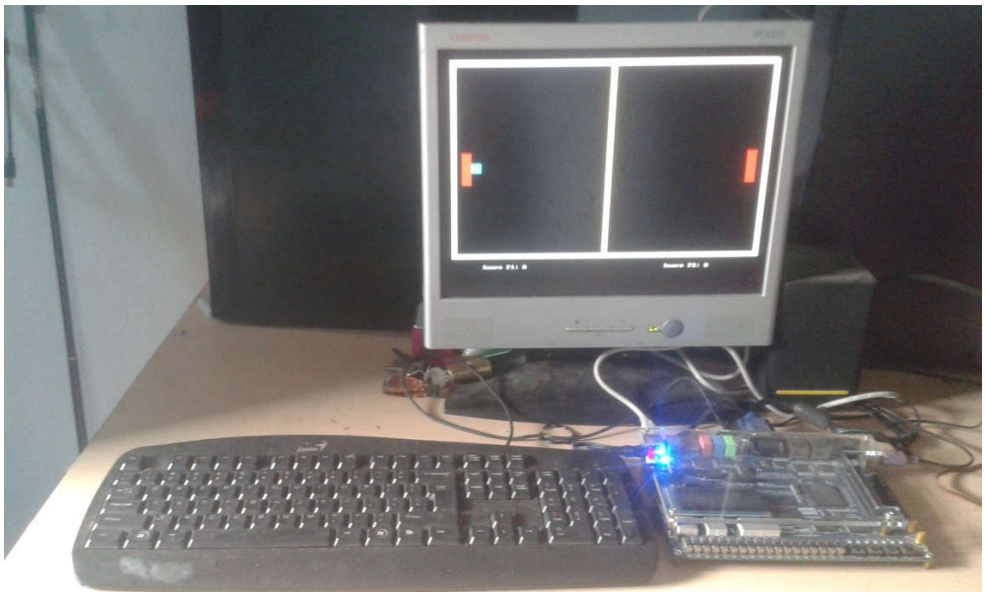
Khi được thực thi, chương trình con kiểm tra kết nối phần cứng tới các hệ thống VGA. Nếu kết nối thành công chương trình sẽ đưa ra thông báo có thể truyền nhận dữ liệu, ngược lại chương trình sẽ đưa ra thông báo lỗi kết nối.

Chương 4

KẾT QUẢ THỰC HIỆN

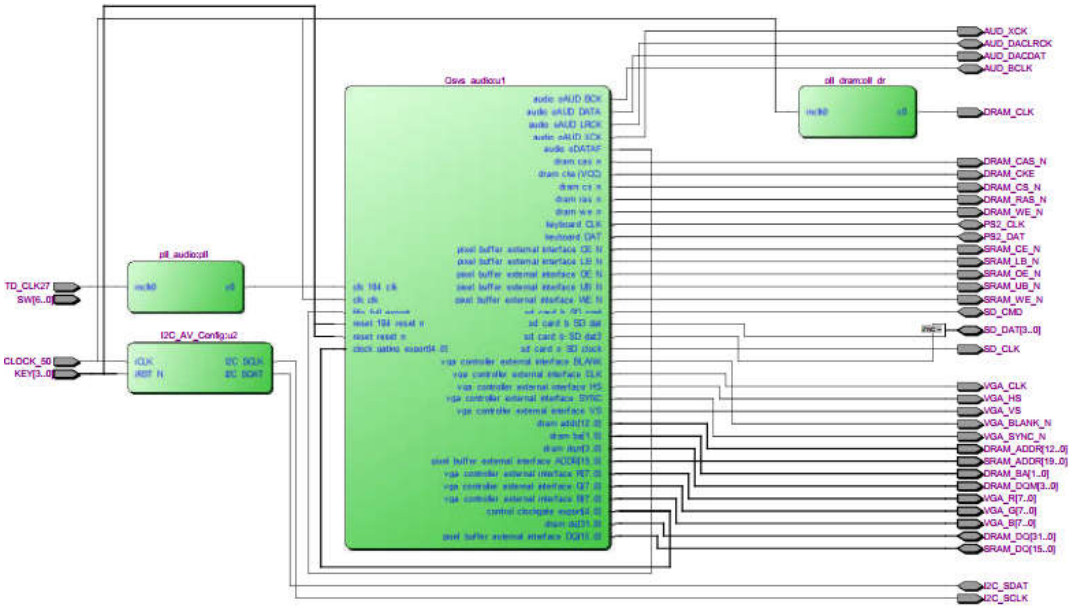
4.1 KẾT QUẢ XÂY DỰNG HỆ THỐNG

❖ *Phần cứng*



Hình 4. 1: Hệ thống nhúng thiết kế sử dụng Kit Altera DE2-115

Phần cứng hệ thống đã xây dựng hoàn chỉnh bao gồm bàn phím máy tính chuẩn PS/2, kit nhúng FPGA Altera DE2-115, màn hình LCD, loa và SD Card.



Hình 4. 2: Sơ đồ phần cứng ở mức netlist sau khi tổng hợp

Hình 4.2 là kết quả phần cứng của hệ thống SoC sau khi đã tổng hợp có thể được xem bằng dạng sơ đồ khối hoặc ở mức netlist thông qua công cụ trên Quartus. Người thiết kế có thể xem lại tổng quan toàn bộ hệ thống thông qua đó có thể có sự điều chỉnh cần thiết cho hệ thống hoặc kiểm tra lỗi kết nối các module hay tín hiệu không mong muốn.

System Contents | Address Map | Clock Settings | Project Settings | Instance Parameters | System Inspector | HDL Example | Generation

Use	C...	Name	Description	E...	Clock	Base	End	IRQ	Op
✓		clk_0	Clock Source						
✓		nios2_qsys_0	Nios II Processor		clk_0	0x0820_2800	0x0820_2fff		
✓		Audio_0	Audio_v1		multiple	0x0820_3458	0x0820_3459		
✓		sdram	SDRAM Controller		clk_0	0x0000_0000	0x07ff_ffff		
✓		sd_card	SD Card Interface		clock_gatin...	0x0820_3000	0x0820_33ff		
✓		fifo_full	PIO (Parallel I/O)		clk_0	0x0820_3430	0x0820_343f	0	
✓		jtag_uart_0	JTAG UART		clk_0	0x0820_3450	0x0820_3457	1	
✓		clk_18_4	Clock Source						
✓		character_buffer_wit...	Character Buffer for VGA Display		control_vga	multiple	multiple		
✓		pixel_buffer	SRAM/SSRAM Controller		clock_gatin...	0x0800_0000	0x081f_ffff		
✓		pixel_buffer_dma	Pixel Buffer DMA Controller		control_vga	0x0820_3420	0x0820_342f		
✓		rgb_resampler	RGB Resampler		control_vga				
✓		scaler	Scaler		control_vga				
✓		alpha_blender	Alpha Blender		control_vga				
✓		dual_clock	Dual-Clock FIFO		multiple				
✓		vga_controller	VGA Controller		clock_gatin...				
✓		up_clocks_0	Clock Signals for DE-series Board Peri...		clk_0				
✓		keyboard	PS2 Controller		clk_0	0x0820_3440	0x0820_3447	2	
✓		control_clockgate	PIO (Parallel I/O)		clk_0	0x0820_3400	0x0820_341f		
✓		clock_gating	clock_gating		multiple				
✓		control_vga	Clock Source						

Hình 4. 3: Hệ thống thiết kế trên công cụ Qsys

Hình 4.3 là kết quả hệ thống SoC được xây dựng trên công cụ Qsys trên phần mềm Quartus. Các thành phần hệ thống bao gồm các lõi IP được Altera cung cấp và các IP được người xây dựng hệ thống thiết kế. Các thành phần được kết nối thông qua việc cấu hình các IP, cung cấp các tín hiệu cần thiết để kết nối hệ thống dưới sự điều khiển, giám sát của lõi vi xử lý 32-bit Nios II.

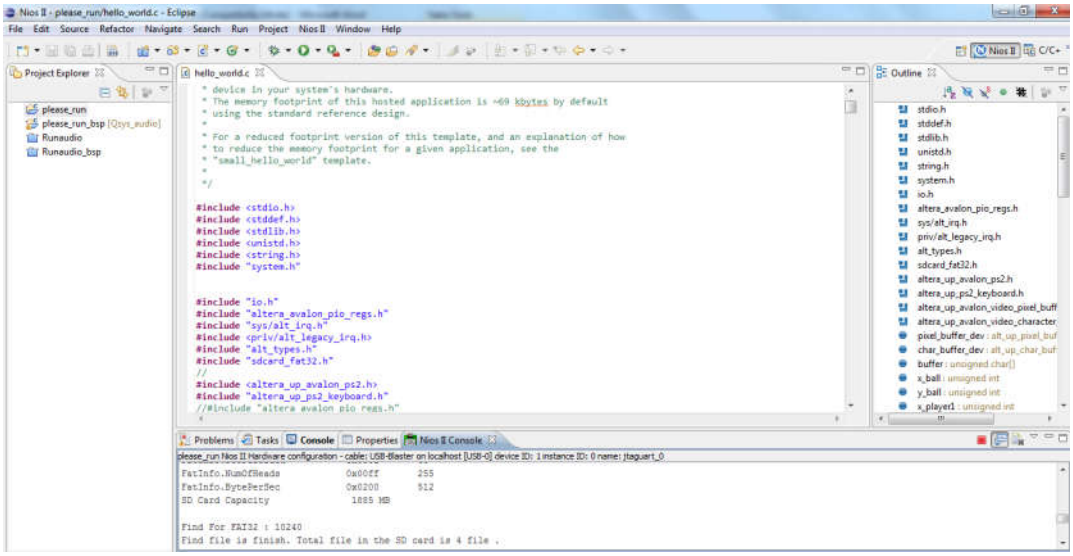
Bảng 2: Bảng các thông số phần cứng hệ thống sau khi tổng hợp

Điệp áp cấp cho chip		1.2V
Điện áp cho I/O		3.3V
Nhiệt độ hoạt động trên chip		$0\text{-}85^0$C
Total logic elements(LEs)		8182/114.480 (chiếm 7% tổng tài nguyên trên chip)
	Total combinational functions	6619/114.480 (6%)
	Dedicated logic registers	5888/114.480 (5%)
Total registers		6007
Total pins		154/529 (29%)
Total memory bits		160.064/3.981.312(4%)
Total PLLs		3/4 (75%)

Các dữ liệu trong bảng 2 được ghi lại thông qua quá trình tổng hợp phần cứng ở mức RTL. Sau quá trình tổng hợp, phần cứng được nạp xuống chip FPGA Cyclone-IV-E trên board DE2-115.

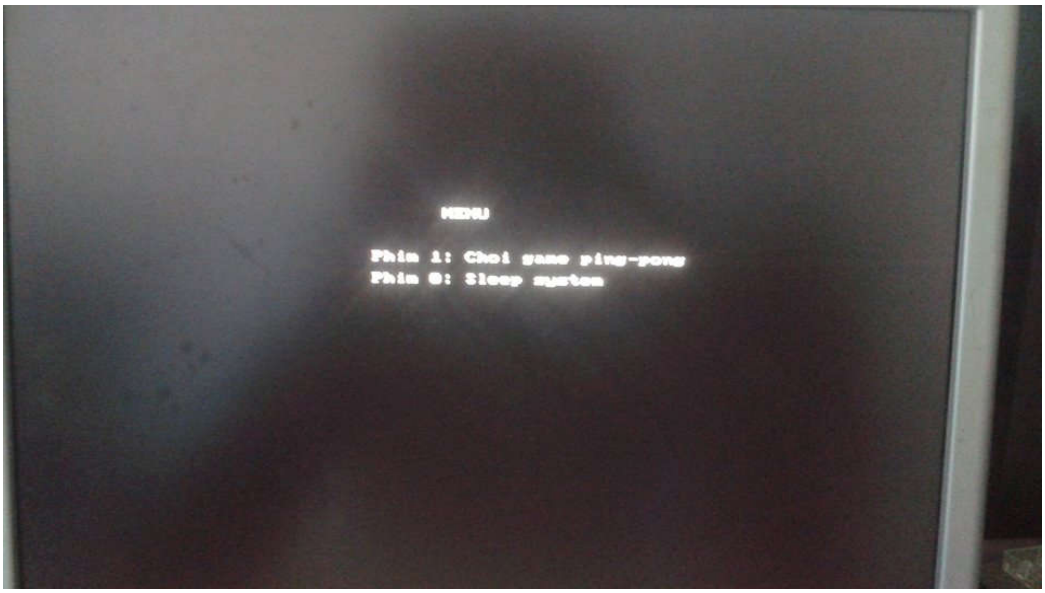
❖ *Phần mềm*

Quá trình xây dựng phần mềm được xây dựng hoàn toàn trên phần mềm Nios II Software Build Tool for Eclipse, phần mềm cho hệ thống được xây dựng bằng ngôn ngữ C và được biên dịch và nạp vào hệ thống phần cứng thông qua RAM trên chip.



Hình 4. 4: Lập trình phần mềm cho hệ thống phần cứng

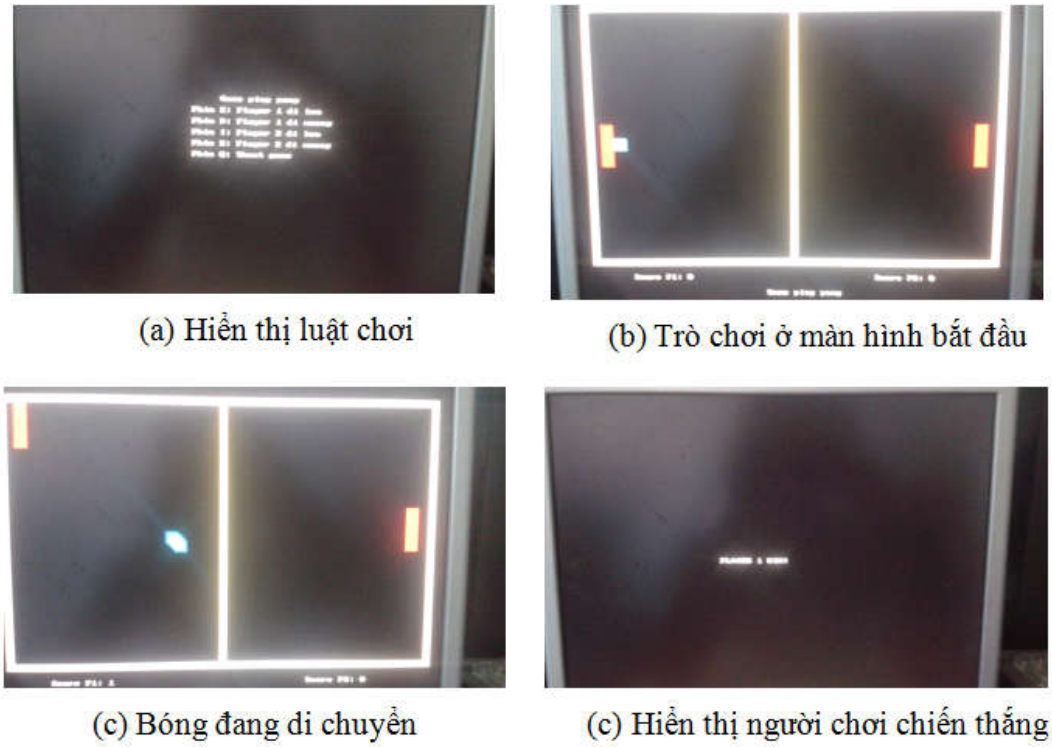
Hình 4.4 là kết quả chương trình phần mềm được viết bằng ngôn ngữ C và debug chương trình phần mềm thông qua Nios II Console được kết nối với phần cứng hệ thống nhờ IP Jtag Uart đã được thiết kế trong hệ thống.



Hình 4. 5: Thao tác với phần mềm hoạt động trên phần cứng được thiết kế

Trong hình 4.5 là kết quả chương trình hoạt động trên hệ thống được hiển thị thông qua màn hình LCD.

❖ *Ứng dụng trò chơi ping-pong*



(a) Hiển thị luật chơi (b) Trò chơi ở màn hình bắt đầu

(c) Bóng đang di chuyển (c) Hiển thị người chơi chiến thắng

Hình 4. 6: (a), (b), (c), (d) Các thao tác với ứng dụng trò chơi ping-pong

Hình 4.6 là kết quả các thao tác với trò chơi được điều khiển bằng bàn phím và hiển thị trên màn hình LCD.

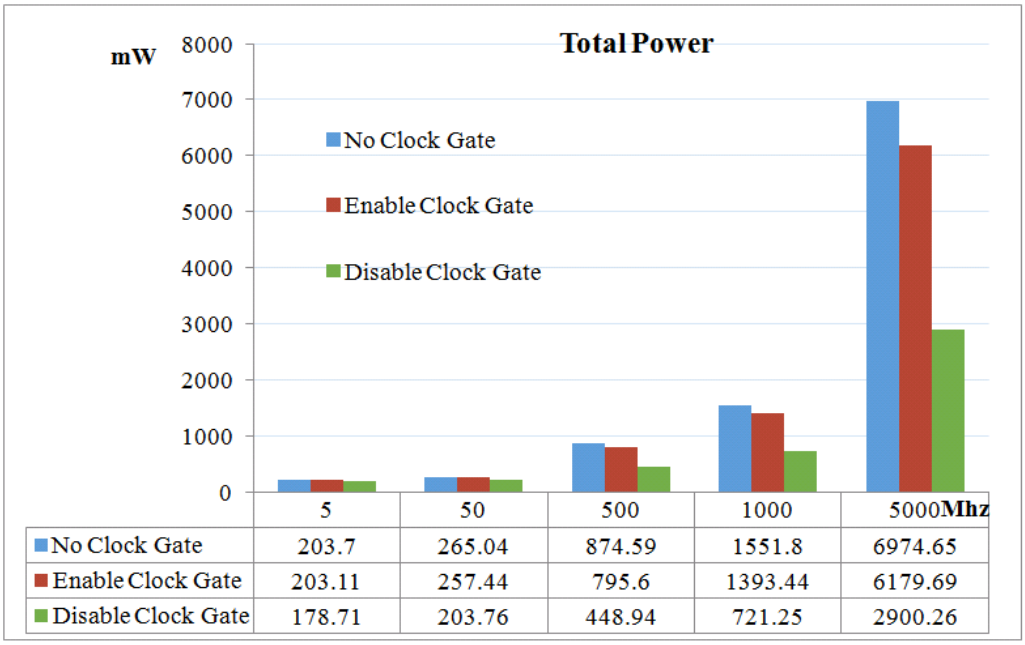
4.2 KẾT QUẢ KHI DÙNG CLOCK GATING CHO HỆ THỐNG

Kết quả hoạt động của hệ thống trên thực tế đã đáp ứng yêu cầu đề tài khi thiết kế clock gating nhằm giảm thiểu năng lượng tiêu tốn cho hệ thống khi hệ thống ở chế độ nghỉ. Hệ thống nhúng trên kit DE2-115 hoạt động ổn định, đáp ứng các yêu về phát nhạc từ SD Card ra loa, người dùng có thể điều khiển hệ thống bằng thao tác trên bàn phím chuẩn PS/2, tương tác qua màn hình LCD hoặc có thể reset hệ thống bằng nút reset nằm trên Kit. Các thông số về tiêu tốn tài nguyên phần cứng, nhiệt độ hoạt động, điện áp cung cấp cho toàn hệ thống trong trường hợp thiết kế có clock gating và không có clock gate được trình bày trong bảng 3.

Bảng 3: So sánh các thông số tài nguyên phần cứng khi thiết kế

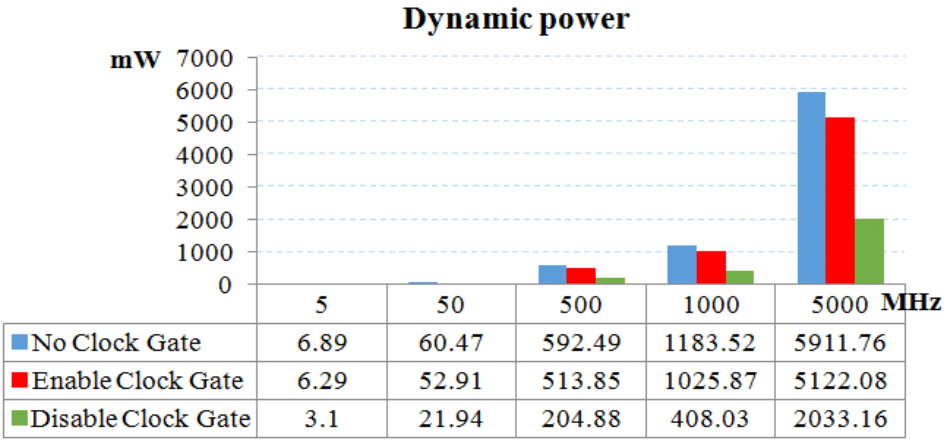
Các thông số hệ thống	Tài nguyên khi thiết kế clock gating.	Tài nguyên khi không thiết kế clock gating.
Total logic elements(LEs)	8182/114.480 (7%)	8133/114.480 (7%)
Total combinational functions	6619/114.480 (6%)	6580/114.480 (6%)
Dedicated logic registers	5888/114.480 (5%)	5875/114.480 (4%)
Total registers	6007	5994
Total pins	154/529 (29%)	154/529 (29%)
Total memory bits	160.064/3.981.312(4%)	160.064/3.981.312(4%)
Total PLLs	3/4 (75%)	3/4 (75%)
Điện áp trên chip	1.2V	
Điệp áp cho các I/O	3.3V	
Nhiệt độ hoạt động	Từ 0^0C - 85^0C	
Công nghệ chip	60-nm	

Các thông số trên bảng 3 cho thấy rằng thiết kế khi có clock gating chỉ chiếm thêm khoảng 60 LEs so với thiết kế không có clock gating. Thiết kế có clock gating chỉ tăng thêm một phần nhỏ diện tích nhưng có thể giảm tiết kiệm phần lớn năng lượng tiêu thụ cho hệ thống. Ngoài ra các công nghệ trên chip như điện áp cấp cho các mạch trên chip chỉ là 1.2V sẽ giúp cho hệ thống ít tiêu tốn về mặt tổng năng lượng trên chip.



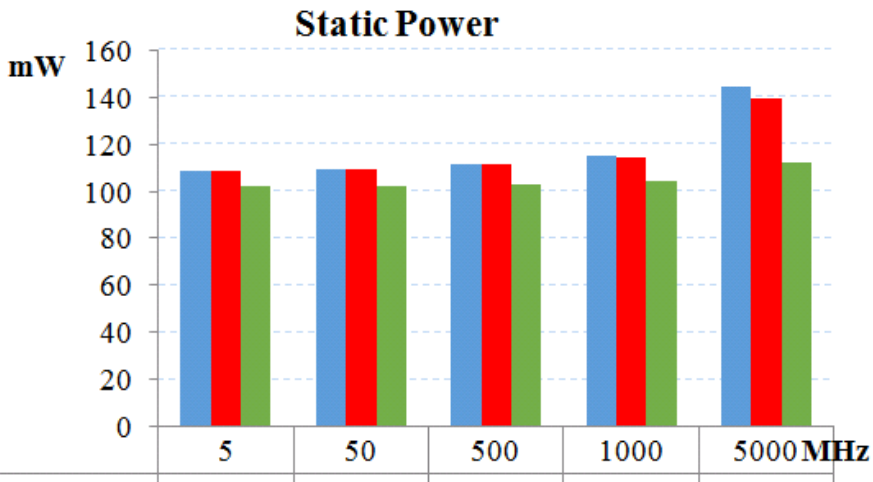
Hình 4. 7: So sánh mức năng lượng tiêu tốn giữa các thiết kế

Có thể thấy trong hình 4.7 rằng mức năng lượng tiêu tốn cho chip với thiết kế có clock gating và vô hiệu hóa clock gating đều có mức năng lượng tiêu tốn nhỏ hơn so với không có clock gating. Ở trường hợp khi vô hiệu hóa clock gate năng lượng có thể giảm tới 58%, 53%, 48%, 24%, 13% tương ứng với các tần số 5GHz, 1GHz, 0.5GHz, 50MHz, 5MHz so với khi không thiết kế clock gating.



Hình 4. 8: So sánh công suất tiêu thụ động giữa các thiết kế

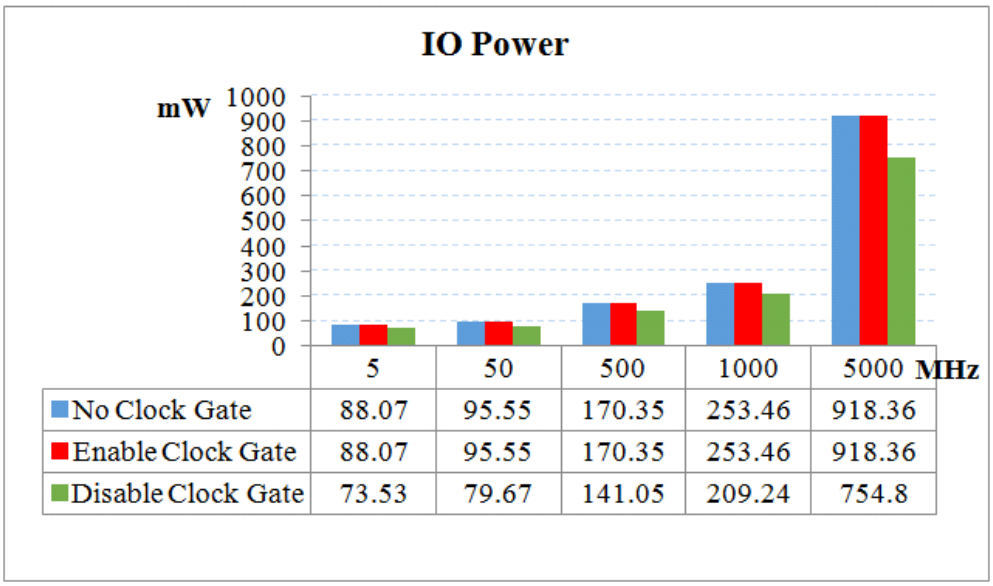
Công suất tiêu thụ động được tính trên sự chuyển đổi logic giữa các thành phần trên chip ví dụ như sự chuyển đổi điện áp từ mức cao xuống thức thấp. Ở các tần số hoạt động càng cao thì thiết kế với clock gate càng trở nên có ích cho hệ thống. Cụ thể với thiết kế clock gate khi mạch ở trạng thái nghỉ có thể giảm thiểu được 66% năng lượng so với khi không thiết kế clock gate.



	5	50	500	1000	5000
No Clock Gate	108.75	109.02	111.75	114.83	144.53
Enable Clock Gate	108.75	108.99	111.4	114.1	139.25
Disable Clock Gate	102.07	102.16	103.01	103.98	112.29

Hình 4. 9: Công suất tiêu thụ tĩnh giữa các thiết kế

Công suất tĩnh là thành phần nhận ít sự tác động về năng lượng trong các thiết kế khác nhau, công suất này có mức độ tăng thấp về năng lượng khi tần số hoạt động của hệ thống thay đổi từ thấp lên cao. Cụ thể ở ở thiết kế có clock gate giảm khoảng 23%, 10%, 8%, 7%, 6% tương ứng với các tần số 5GHz, 1GHz, 500MHz, 50MHz và 5Hz.



	5	50	500	1000	5000
No Clock Gate	88.07	95.55	170.35	253.46	918.36
Enable Clock Gate	88.07	95.55	170.35	253.46	918.36
Disable Clock Gate	73.53	79.67	141.05	209.24	754.8

Hình 4. 10: Công suất tiêu thụ các thành phần I/O

Một thành phần tiêu tốn công suất trong thiết kế nữa đó là công suất các thành phần I/O. Công suất tính dựa trên các mức điện áp được cấp cho các thành phần ngõ vào, ra, các tụ hay trở kháng của các chân kết nối đến chip. Thành phần công suất I/O chiếm một phần tỉ lệ trong tiêu tốn năng lượng của chip. Ở thiết kế có clock gate có thể giảm khoảng 18% năng lượng cho hệ thống so với thiết kế không có clock gate với các mức tần số khác nhau.

Chương 5

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1 KẾT LUẬN

Luận văn tốt nghiệp đã hoàn thành đúng theo các yêu cầu đặt ra. Xây dựng thành công hệ thống nhúng trên kit FPGA. Kết hợp kiến thức đã tìm hiểu về xây dựng một hệ thống SoC, lập trình phần cứng bằng ngôn ngữ verilog, phần mềm bằng ngôn ngữ C để xây dựng, hoàn thiện mô hình. Hệ thống nhúng trên kit FPGA đảm bảo các yêu cầu về giao tiếp với các thiết bị Loa, SD Card, bàn phím PS/2, màn hình LCD và kit nhúng FPGA Altera DE2-115. Các chương trình nhúng hoạt động ổn định trên hệ thống đã được thiết kế.

Ưu điểm :

- Hệ thống hoạt động ổn định.
- Có thể áp dụng vào thực tiễn.

Nhược điểm :

- Phần cứng chưa tối ưu hoàn toàn.
- Chưa có nhiều ứng dụng trên phần mềm.
- Cần cải tiến hệ thống clock gating.

5.2 HƯỚNG PHÁT TRIỂN DỀ TÀI

Thay đổi lõi xử lý Nios II bằng các lõi được phát triển rộng rãi trên thư viện OpenCore cũng như thay đổi các IP có sẵn được cung cấp theo phần mềm Quartus nhằm giới hạn về vấn đề bản quyền của các lõi IP.

Cải thiện hệ thống clock gating nhằm tối ưu hơn việc tiêu tốn năng lượng cho hệ thống hoặc áp dụng các phương pháp năng lượng khác.

Tối ưu lại hệ thống phần cứng, hạn chế tiêu tốn tài nguyên phần cứng qua đó có thể giảm năng lượng tiêu hao.

Tối ưu hệ thống phần mềm nhằm đưa hệ thống hoạt động ổn định hơn, xây dựng nhiều ứng dụng hơn cho hệ thống.

TÀI LIỆU THAM KHẢO

[1] R. Saleh, S. Wilton, S. Mirabbasi, A. Hu, M. Greenstreet, G. Lemieux, P. P. Pande, C. Grecu, A. Ivanov, "System-on-Chip: Reuse and Integration," *Proceedings of the IEEE,* vol. 94, no. 6, pp. 1050 - 1069, June 2006.

[2] Pong P.Chu, Embedded SoPC Design with Nios II Processor and VHDL Examples, A John Wiley & Sons, Inc, August 2011.

[3] Roland Szabó, Aurel Gontean, "Pong game on FPGA with CRT or LCD display and push button controls," in *Proceedings of the 2014 Federated Conference on Computer Science and Information Systems*, 2014.

[4] K. Liu, Y. Yang and Y. Zhu, "Tetris game design based on the FPGA," in *International Conference on Consumer Electronics, Communications and Networks*, 2012.

[5] Altera, "DE2_115_User_Manual," [Online]. Available: ftp://ftp.altera.com/up/pub/Altera_Material/Boards/DE2-115/DE2_115_User_Manual.pdf. [Accessed 10 April 2016].

[6] N. H. Quân, H. N. T. Hoàng, N. P. Quốc, Đ. N. Quỳnh, T. K. Cường, L. Phúc and N. D. Thái, Verilog HDL và ứng dụng, TP. Hồ Chí Minh: Nhà xuất bản Đại Học Quốc Gia, 2011.

[7] B. Pandey, J. Yadav, J. Kumar and R. Kumar, "Clock Gating Aware Low Power Global Reset ALU and Implementation on 28nm FPGA," in *International Conference on Computational Intelligence and Communication Networks*, 5th 2013.

[8] T. V. Quang, "Lab 2 - PS2 Controller," ICDREC, TP. Hồ Chí Minh, 2015.

[9] A. Chapweske, "The PS/2 Mouse/Keyboard Protocol," [Online]. Available: http://www.computer-engineering.org/ps2protocol/. [Accessed 1 April 2016].

[10] T. V. Quang, "Lab 3 - SD Card Controller," ICDREC, TP. Hồ Chí Minh, 2015.

[11] T. V. Quang, "Lab 5 - Audio CODEC," ICDREC, TP. Hồ Chí Minh, 2015.

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT THÀNH PHỐ HỒ CHÍ MINH

www.hcmute.edu.vn