

THIẾT KẾ VÀ THI CÔNG BỘ TRUYỀN NHẬN UART ĐA KÊNH DỰA TRÊN CÔNG NGHỆ FPGA

DESIGN AND FPGA-BASED IMPLEMENTATION OF A MULTICHANNEL UART MODULE

Phạm Thái Ngọc, Hồ Ngọc Bảo, Đỗ Duy Tân, Trương Quang Phúc
 Trường Đại học Sư phạm Kỹ thuật TP.HCM, Việt Nam

Ngày tòa soạn nhận bài 9/8/2020, ngày phản biện đánh giá 22/8/2020, ngày chấp nhận đăng 01/9/2020.

TÓM TẮT

UART (Universal Asynchronous Receiver and Transmitter) là chuẩn truyền nối tiếp bất đồng bộ khá phổ biến hiện nay. Mặc dù tốc độ truyền dữ liệu không quá cao nhưng với ưu điểm đơn giản, dễ thực hiện và tiêu hao công suất thấp, UART được sử dụng trong các module không yêu cầu tốc độ giao tiếp cao như module SIM, Bluetooth, GPS. Tuy nhiên, việc giao tiếp với nhiều ngoại vi tốc độ thấp có thể làm giảm hiệu suất sử dụng bus dữ liệu của hệ thống và hiệu suất hoạt động của vi xử lý. Do đó, trong bài báo này, chúng tôi xin trình bày một thiết kế module UART đa kênh được thiết kế để sử dụng tốt hơn bus dữ liệu theo chuẩn APB (Advanced Peripheral Bus) và hỗ trợ được nhiều tốc độ và cấu trúc khung dữ liệu truyền UART khác nhau. Sau đó, chức năng của thiết kế UART được đánh giá thông qua kết quả mô phỏng và thực thi trên kit FPGA (Field Programmable Gate Arrays).

Từ khóa: UART đa kênh; APB; thiết kế dựa trên FPGA; phần cứng; mô phỏng.

ABSTRACT

Universal Asynchronous Receiver and Transmitter (UART) is a popular asynchronous serial transmission standard. Although the data transfer rate is not too high, with the advantages of simplicity, ease of implementation and low power consumption, UART has been frequently used in modules that do not require high communication speed such as SIM module, Bluetooth, GPS, etc. However, communication with low-speed peripherals can reduce the efficiency of data bus and processor performance. Therefore, in this paper, we present a multichannel UART design to better utilize the APB (Advanced Peripheral Bus) standard data bus and to support multiple transmission data frames with different rates. Then, the performance of the designed multichannel UART module is evaluated through simulations and FPGA-based implementation (Field Programmable Gate Arrays).

Keywords: Multichannel UART; APB; FPGA-based design; hardware; testbench.

1. GIỚI THIỆU

Để tối ưu việc giao tiếp với các chuẩn bus lớn của các hệ thống hiện đại, các thiết kế bộ điều khiển truyền nhận UART có khả năng quản lý nhiều kênh và sử dụng bus dữ liệu 32 bit đã được đưa ra nhưng lại dựa trên các bus dữ liệu 32 bit cũ của Motorola PowerPC series [1]. Thiết kế UART một kênh bằng ngôn ngữ miêu tả phần cứng cũng đã được trình bày qua các báo cáo [2]-[7], tuy nhiên thiết kế còn đơn giản không bao

hàm nhiều cấu hình, tốc độ truyền và kiểm tra bit parity (chẵn/lẻ), chỉ được chọn ở một cấu hình cố định và tốc độ truyền cao nhất có thể. Để cải thiện hạn chế của các thiết kế đã có, bài báo này trình bày thiết kế một module UART đa kênh hoàn chỉnh sử dụng giao tiếp bus mới hơn là AMBA 3 APB (Advanced Microcontroller Bus Architecture 3 - Advanced Peripheral Bus) của ARM [8]-[9], sử dụng phần cứng tạo ngắt yêu cầu đọc đơn giản hơn, đặt độ phức tạp lên phần mềm nhằm giảm thiểu tài nguyên logic so với thiết

kế cũ. Hình 1 minh họa hệ thống System-On-Chip (SOC) sử dụng bus AMBA và module Multichannel UART. Cụ thể, module truyền nhận đơn lẻ vẫn duy trì kỹ thuật cải thiện tốc độ truyền đồng thời bổ sung thêm khả năng hỗ trợ nhiều tốc độ truyền, cấu hình truyền bao gồm việc kiểm tra lỗi parity và thay đổi độ rộng bit stop [10]. Thiết kế mở rộng thêm bộ đồng bộ tín hiệu để có thể thực hiện các giao tiếp bus bất đồng bộ. Thiết kế được đánh giá qua mô phỏng và kiểm tra thực tế trên FPGA Cyclone II - EP2C5T144C8 [12].

2. THIẾT KẾ CHI TIẾT HỆ THỐNG

Các module truyền nhận được mở rộng hoạt động dựa trên các tín hiệu cấu hình, các bộ đệm FIFO (First-In First-Out) [13], các tập thanh ghi để lưu dữ liệu cấu hình và đồng bộ tín hiệu, cuối cùng là module giao tiếp với bus hệ thống. Trong bài báo này, chúng tôi xem xét thiết kế UART 4 kênh như một ví dụ minh họa. Sơ đồ khối của một module UART đa kênh được thể hiện trong Hình 2.

2.1. Module Baud_Rate generator

Bộ truyền nhận UART được thiết kế phổ biến dựa trên phương thức lấy mẫu nhiều lần cho một bit dữ liệu. Thông thường một bit dữ liệu UART có thể được lấy mẫu 16 lần và

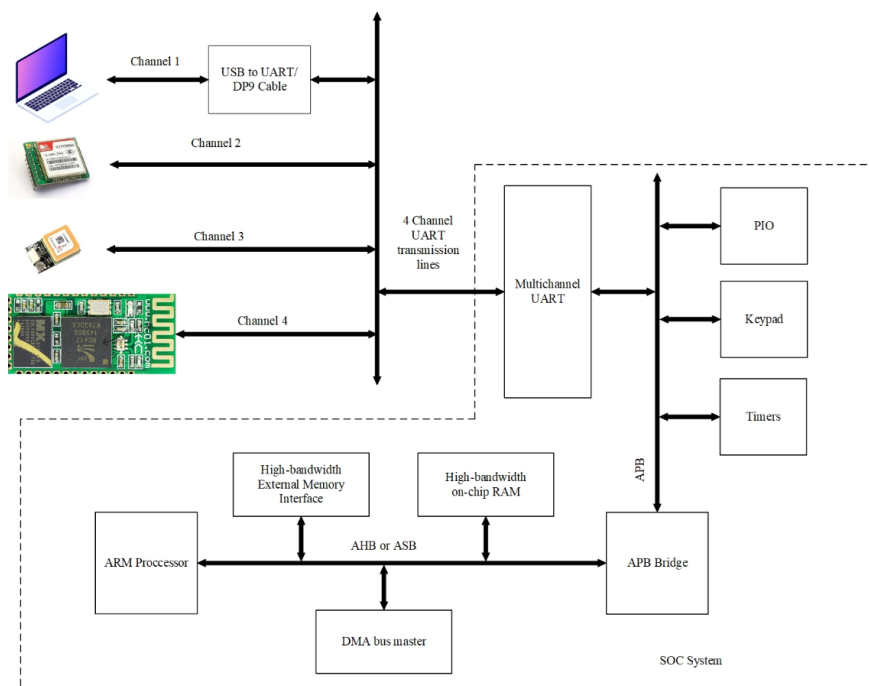
chọn điểm giữa của bit dữ liệu hay lần lấy mẫu thứ 8 để lấy giá trị của bit dữ liệu đó. Thời gian lấy mẫu sẽ phụ thuộc vào tín hiệu s_tick tạo bởi khối Baud_Rate Generator, do đó chỉ cần thay đổi s_tick thì tốc độ truyền sẽ được thay đổi.

2.2. Module Register File

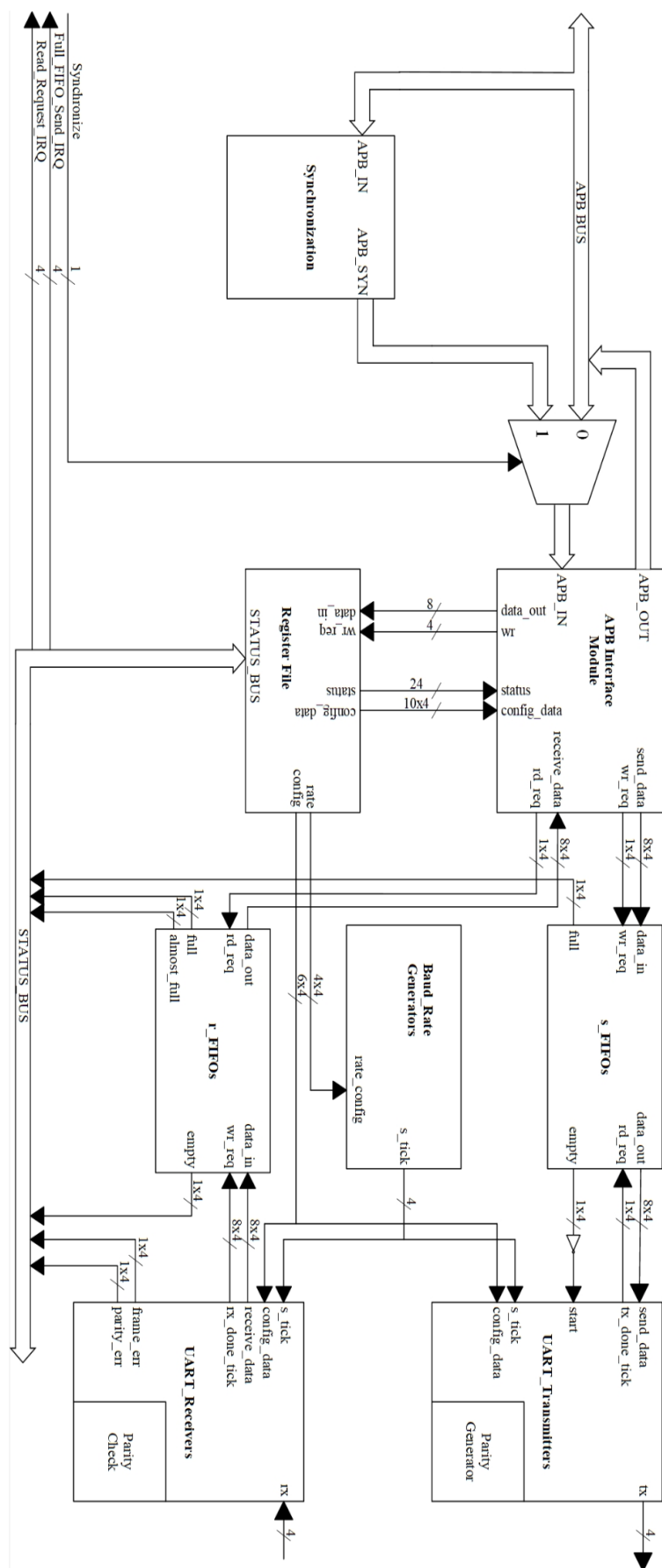
Khối Register Files gồm các thanh ghi lưu giá trị cấu hình và trạng thái của các kênh truyền UART. Mỗi kênh truyền cần 10 bit để cấu hình, 6 bit, ký hiệu Frame_Config, để cấu hình frame truyền nhận và 4 bit, ký hiệu là Rate_Config, để cấu hình tốc độ baud. Khối APB Interface sẽ giao tiếp trực tiếp với với tập thanh ghi này để cấu hình các kênh truyền.

2.3. Các bộ đệm sFIFO và rFIFO

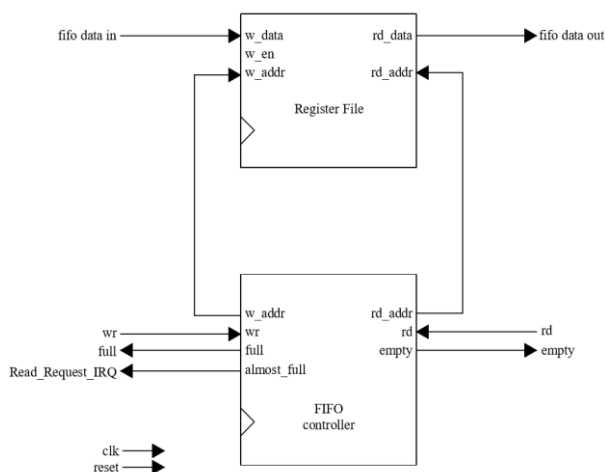
sFIFO là bộ đệm dữ liệu để gửi đi và rFIFO là bộ đệm dữ liệu nhận được. Bộ đệm FIFO có nhiều cách thiết kế, có thể bất đồng bộ xung Clock đọc ghi [14], có thể nhiều port đọc ghi, thiết kế này chỉ sử dụng một bộ đệm FIFO đơn giản nhất (minh họa trong Hình 3). Bộ đếm địa chỉ nhị phân sẽ sử dụng thêm 1 bit để phân biệt đầy và trống giữa hai con trỏ địa chỉ đọc và ghi.



Hình 1. Hệ thống SOC sử dụng bus AMBA và module Multichannel UART.



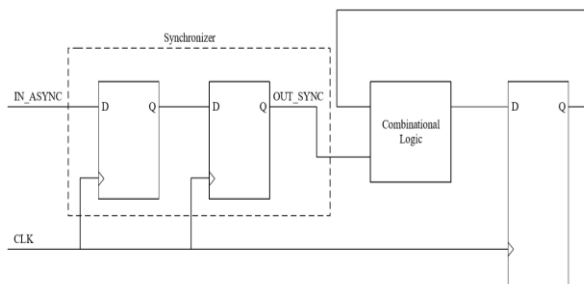
Hình 2. Sơ đồ khối của hệ thống.



Hình 3. Cấu tạo bộ đệm FIFO.

2.4. Module Synchronization

Việc hai thiết bị số không đồng bộ giao tiếp với nhau có thể dẫn đến vi phạm các yếu tố định thời của các Flip Flop ở hai thiết bị, khiến các thiết bị này rơi vào trạng thái Metastable [15]. Chúng ta có thể thiết kế các mạch đồng bộ gồm các Flip Flop ghép nối tiếp với nhau để tái đồng bộ tín hiệu.



Hình 4. Mạch đồng bộ dùng 2 Flip Flop.

2.5. Module UART Transmitter

Một module UART Transmitter riêng lẻ chỉ truyền được dữ liệu cho 1 kênh, do đó có tổng cộng 4 module UART Transmitter cho 4 kênh truyền. Khối hoạt động dựa trên tín hiệu cấu hình từ tập thanh ghi Register File, tín hiệu s_tick từ bộ Baud Rate Generator và tín hiệu Parity của bộ Parity Generator. Bộ Baud Rate Generator và tập thanh ghi Register File đã được trình bày ở trên, còn module Parity Generator thực chất là một mạch XOR tín hiệu ngõ vào lại và tạo ra bit Parity chẵn, do hệ thống hoạt động với số bit dữ liệu có thể điều chỉnh được và có hỗ trợ cả Parity lẻ nên

bộ Parity Generator còn cần thêm mạch tổ hợp quyết định ngõ ra [10].

2.6. Module UART Receiver

Module UART Receiver cũng hoạt động dựa trên tín hiệu cấu hình từ tập thanh ghi Register File, tín hiệu s_tick và tín hiệu Parity. Tuy nhiên ngõ vào tín hiệu UART là một ngõ vào bất đồng bộ, do đó cần tái đồng bộ ngõ như được thực hiện ở module Synchronization, tín hiệu rx sẽ được tái đồng bộ qua 2 Flip Flop. Bộ Parity Check cũng gồm một mạch XOR tín hiệu nhưng có mạch tổ hợp lựa chọn tín hiệu ra đơn giản hơn bộ Parity Generator, do khi bắt đầu nhận dữ liệu thì khối UART Receiver sẽ reset thanh ghi chứa dữ liệu nhận về mức logic 0 và XOR với bit 0 thì không thay đổi kết quả, nên bộ Parity Check không cần lựa chọn số lượng bit XOR mà dùng một mạch XOR 8 bit cố định.

Do giới hạn số trang, chúng tôi không trình bày chi tiết lưu đồ ASM (Algorithmic State machine) cho UART Receiver và Transmitter.

2.7. Module APB Interface

Module APB Interface có chức năng giao tiếp với bus APB đồng thời điều khiển ghi/đọc dữ liệu. Giao tiếp APB theo chuẩn AMBA 3 APB sẽ có thêm tín hiệu PREADY để thực hiện giao tiếp có trạng thái chờ cho hai hệ thống không đồng bộ. Module có mô hình trạng thái hữu hạn như [8]-[9].

2.8. Mô tả thanh ghi

Các thanh ghi được định nghĩa trong thiết kế được miêu tả qua Bảng 1. Cấu hình chi tiết và chức năng của các thanh ghi trên được miêu tả qua Bảng 2.

Bảng 1. Các thanh ghi ứng với địa chỉ PADDR và kiểu truy cập

| PADDR [3:0] | Kiểu truy cập | Tên thanh ghi | Chức năng |
|-------------|---------------|-----------------|-----------------------------|
| 0000 | Đọc/Ghi | Baud_Rate_Ch1_2 | Cấu hình tốc độ kênh 1 và 2 |
| 0001 | Đọc/Ghi | Baud_Rate_Ch3_4 | Cấu hình tốc độ kênh 3 và 4 |
| 0010 | Đọc/Ghi | Frame_Config_1 | Cấu hình frame kênh 1 |

| PADDR [3:0] | Kiểu truy cập | Tên thanh ghi | Chức năng |
|-------------|---------------|----------------|-------------------------------|
| 0011 | Đọc/Ghi | Frame_Config_2 | Cấu hình frame kênh 2 |
| 0100 | Đọc/Ghi | Frame_Config_3 | Cấu hình frame kênh 3 |
| 0101 | Đọc/Ghi | Frame_Config_4 | Cấu hình frame kênh 4 |
| 0110 | Chỉ Ghi | Write_1 | Ghi dữ liệu truyền đi kênh 1 |
| 0111 | Chỉ Ghi | Write_2 | Ghi dữ liệu truyền đi kênh 2 |
| 1000 | Chỉ Ghi | Write_3 | Ghi dữ liệu truyền đi kênh 3 |
| 1001 | Chỉ Ghi | Write_4 | Ghi dữ liệu truyền đi kênh 4 |
| 1010 | Chỉ Đọc | Status | Đọc trạng thái bộ truyền nhận |
| 1011 | Chỉ Đọc | Read_1 | Đọc dữ liệu kênh 1 |
| 1100 | Chỉ Đọc | Read_2 | Đọc dữ liệu kênh 2 |
| 1101 | Chỉ Đọc | Read_3 | Đọc dữ liệu kênh 3 |
| 1110 | Chỉ Đọc | Read_4 | Đọc dữ liệu kênh 4 |
| 1111 | Chỉ Đọc | Read_All | Đọc dữ liệu cả 4 kênh |

Bảng 2. Cấu hình và chức năng của các thanh ghi trong thiết kế

| Tên thanh ghi | Miêu tả |
|-----------------|---|
| Baud_Rate_Ch1_2 | Cấu hình tốc độ truyền nhận kênh 1 và 2. [3:0]: kênh 1. [7:4]: kênh 2. |
| Baud_Rate_Ch3_4 | Cấu hình tốc độ truyền nhận của kênh 3 và 4 [3:0]: kênh 4. [7:4]: kênh 5. |
| Frame_Config_1 | Cấu hình Frame cho kênh truyền 1. [1:0]: Parity. [3:2]: Stop Bit. [5:4]: số Data Bit. [6]: Cho phép ngắt. |
| Frame_Config_2 | Tương tự Frame_Config_1, cấu hình cho kênh 2. |
| Frame_Config_3 | Tương tự Frame_Config_1, cấu hình cho kênh 3. |
| Frame_Config_4 | Tương tự Frame_Config_1, cấu hình cho kênh 4. |
| Write_1 | Chứa dữ liệu sẽ gửi đi qua kênh truyền UART 1. |
| Write_2 | Chứa dữ liệu cho kênh truyền thứ 2. |
| Write_3 | Chứa dữ liệu cho kênh truyền thứ 3. |
| Write_4 | Chứa dữ liệu cho kênh truyền thứ 4. |

| Tên thanh ghi | Miêu tả |
|---------------|--|
| Status | Chứa trạng thái hoạt động của hệ thống. [3:0]: Full_send, thể hiện việc đầy bộ đệm gửi cho 4 kênh truyền, ứng với ngắt Full_FIFO_Send_IRQ. [7:4]: Full_receive, thể hiện việc đầy bộ đệm nhận dữ liệu cho 4 kênh truyền. [11:8]: Frame_err, báo lỗi Frame khi nhận tín hiệu từ kênh truyền cho 4 kênh (bit [8] cho kênh 1) [15:12]: Parity_err, báo lỗi Parity khi nhận dữ liệu từ kênh truyền cho 4 kênh. [19:16]: Data_available, báo hiệu có dữ liệu hợp lệ trong bộ đệm FIFO nhận cho 4 kênh. [23:20]: Rd_req, báo hiệu bộ đệm nhận đã đầy 70%, ứng với ngắt Read_Request_IRQ. |
| Read_1 | Chứa dữ liệu nhận được từ kênh truyền 1. [7:0]: kênh 1. [31:8]: Toàn bộ bit 0. |
| Read_2 | Chứa dữ liệu đọc về của kênh 2. |
| Read_3 | Chứa dữ liệu đọc về của kênh 3. |
| Read_4 | Chứa dữ liệu đọc về của kênh 4. |
| Read_All | Chứa dữ liệu nhận của cả 4 kênh truyền. [7:0]: kênh 1. [15:8]: kênh 2. [23:16]: kênh 3. [31:24]: kênh 4. |

3. ĐÁNH GIÁ CHẤT LƯỢNG CỦA THIẾT KẾ

3.1 Tài nguyên logic sử dụng

Thiết kế với đầy đủ các chức năng được tổng hợp trên FPGA Cyclone II EP2C5T144C8 có tài nguyên logic được sử dụng như tóm tắt trong Hình 5. Tần số hoạt động tối đa của mạch được tổng hợp trên FPGA Cyclone II EP2C5T144C8 là 128.11 MHz (xem Hình 6).

| Flow Summary | |
|------------------------------------|--|
| Flow Status | Successful - Tue Jul 14 20:15:34 2020 |
| Quartus II 64-Bit Version | 12.1 Build 177 11/07/2012 SJ Web Edition |
| Revision Name | UART_APB |
| Top-level Entity Name | Quad_UART_APB |
| Family | Cyclone II |
| Device | EP2C5T144C8 |
| Timing Models | Final |
| Total logic elements | 1,282 / 4,608 (28 %) |
| Total combinational functions | 1,181 / 4,608 (26 %) |
| Dedicated logic registers | 728 / 4,608 (16 %) |
| Total registers | 728 |
| Total pins | 67 / 89 (75 %) |
| Total virtual pins | 0 |
| Total memory bits | 12,288 / 119,808 (10 %) |
| Embedded Multiplier 9-bit elements | 0 / 26 (0 %) |
| Total PLLs | 0 / 2 (0 %) |

Hình 5. Tài nguyên Logic sử dụng trên FPGA.

| Slow Model Fmax Summary | | | | |
|-------------------------|------------|-----------------|------------|------|
| | Fmax | Restricted Fmax | Clock Name | Note |
| 1 | 123.11 MHz | 123.11 MHz | PCLK | |

Hình 6. Tần số tối đa của thiết kế trên FPGA.

3.2 Kết quả đánh giá qua mô phỏng

Thiết kế sẽ được mô phỏng qua 6 Testcase được liệt kê trong Bảng 3, bao hàm hết các tốc độ và cấu hình hỗ trợ. Dữ liệu cấu hình và dữ liệu nhận được cũng được đọc và ghi qua các chu kỳ giao tiếp bus APB (xem Hình 7). Lần lượt 6 Testcase đã được thực hiện mô phỏng. Tất cả Testcase đều được kiểm tra, so sánh và đều có kết quả đúng với lý thuyết. Dựa vào đây có thể kết luận thiết kế có hoạt động tốt trên mô phỏng, đảm bảo việc truyền nhận UART ở nhiều tốc độ và cấu hình khung truyền khác nhau, khả năng giao tiếp theo chuẩn bus APB.

Bảng 3. Bảng tóm tắt các Testcase.

| Testcase | Nội dung (tốc độ truyền (symbol/s) /số bit dữ liệu/ số bit stop/ Parity) |
|------------|---|
| Testcase 1 | Các kênh truyền được cấu hình mặc định: 9600/ 8/ 1/ không parity |
| Testcase 2 | Cấu hình tốc độ của các kênh truyền: - Kênh 1: 600/ 8/ 1/ không parity - Kênh 2: 1200/ 8/ 1/ không parity - Kênh 3: 2400/ 8/ 1/ không parity - Kênh 4: 4800/ 8/ 1/ không parity |
| Testcase 3 | Tương tự Testcase 2 với cấu hình tốc độ - Kênh 1: 9600 symbol/s - Kênh 2: 14400 symbol/s - Kênh 3: 19200 symbol/s - Kênh 4: 28800 symbol/s |
| Testcase 4 | Tương tự Testcase 2 với cấu hình tốc độ - Kênh 1: 38400 symbol/s - Kênh 2: 56000 symbol/s - Kênh 3: 57600 symbol/s - Kênh 4: 115200 symbol/s |
| Testcase 5 | Tương tự Testcase 2 với cấu hình tốc độ - Kênh 1: 128000 symbol/s - Kênh 2: 256000 symbol/s - Kênh 3: 128000 symbol/s - Kênh 4: 256000 symbol/s |
| Testcase 6 | Cấu hình frame khác nhau Cấu hình frame kênh 1: |

| Testcase | Nội dung (tốc độ truyền (symbol/s) /số bit dữ liệu/ số bit stop/ Parity) |
|----------|--|
| | - 9600/ 8/ 1.5/ Parity lẻ Cấu hình frame kênh 2: - 9600/ 7/ 2/ Parity chẵn Cấu hình frame kênh 3: - 9600/ 6/ 1.5/ Parity chẵn Cấu hình frame kênh 4: - 9600/ 5/ 2/ Parity lẻ |

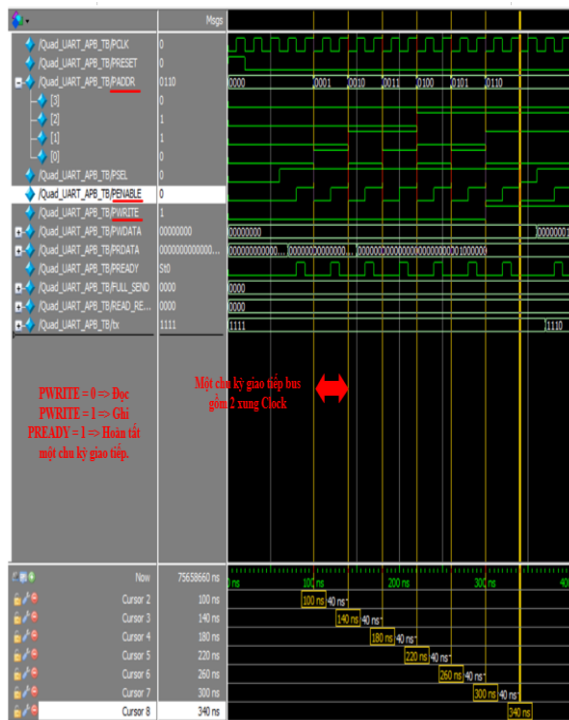
3.3 Kết quả đánh giá thực tế trên FPGA

Để kiểm tra chất lượng module UART được thiết kế khi hoạt động gần với thực tế, chúng tôi dùng kit FPGA như một công cụ hiệu quả, gần với thực tế để đánh giá chất lượng của thiết kế hệ thống số [11]. Kit FPGA được sử dụng là kit Altera Cyclone II với mã cụ thể là EP2C5T144C8 [12]. Tần số Clock chính của hệ thống là 50 MHz.

Chúng tôi cần sử dụng thêm các module có chức năng giao tiếp và hiển thị như được minh họa trong Hình 8 và 9. Để giao tiếp với máy tính và người kiểm tra, FPGA được gắn vào một mạch PCB (Printed Circuit Board) có các nút nhấn và màn hình LCD 16x2, các module USB – UART được dùng để kết nối PCB và máy tính. Thiết kế sẽ được kiểm tra để đảm bảo hoạt động truyền nhận qua các Testcase giống với phần mô phỏng bằng phần mềm Terminal. Sau đó sẽ tiến hành nhận liên tục một chuỗi dữ liệu bằng phần mềm Realterm Serial Capture Program [16] để đánh giá tỷ lệ lỗi và độ ổn định của thiết kế. Phần mềm sẽ truyền file text có dung lượng 1 Mbytes với dữ liệu là chuỗi 0123456789 lặp đi lặp lại. Dựa vào chuỗi tăng dần và lặp lại này module Tester sẽ xác định dữ liệu nhận được là đúng hay sai. Với phần mềm Terminal, sử dụng nút nhấn để gửi dữ liệu lên chậm và không liên tục. Đồng thời dữ liệu ký tự sẽ được gửi đến FPGA và hiển thị trên LCD 16x2. Sau khi đảm bảo được hoạt động của mạch sẽ dùng phần mềm Realterm Serial Capture Program [16] để truyền nhận 1 Mbytes dữ liệu nối tiếp nhằm đánh giá tỷ lệ lỗi và độ ổn định. Kết quả được thống kê qua Bảng 4.

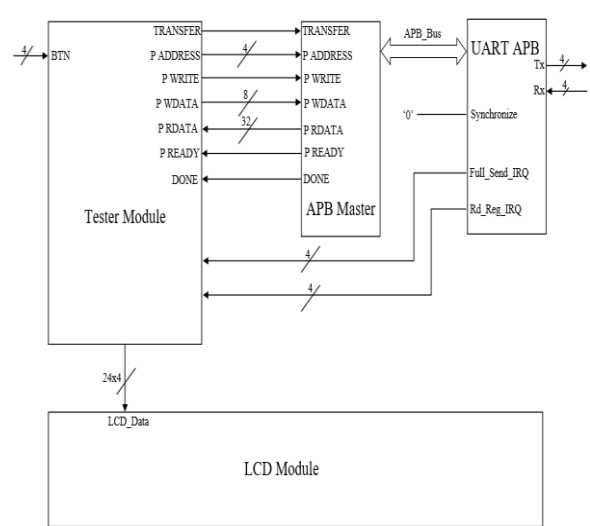
Bảng 4. Kết quả truyền nhận dữ liệu theo chuỗi.

| Baud Rate | Data Bit | Parity | Stop Bit | Error |
|-----------|----------|--------|----------|-------|
| 2400 | 8 | None | 1 | 0 |
| 4800 | 8 | None | 1 | 0 |
| 9600 | 8 | None | 1 | 0 |
| 14400 | 8 | None | 1 | 0 |
| 19200 | 8 | None | 1 | 0 |
| 28800 | 8 | None | 1 | 0 |
| 38400 | 8 | None | 1 | 0 |
| 56000 | 8 | None | 1 | 0 |
| 115200 | 8 | None | 1 | 0 |
| 128000 | 8 | None | 1 | 0 |
| 256000 | 8 | None | 1 | 0 |
| 9600 | 8 | Odd | 2 | 0 |
| 9600 | 8 | Even | 2 | 0 |

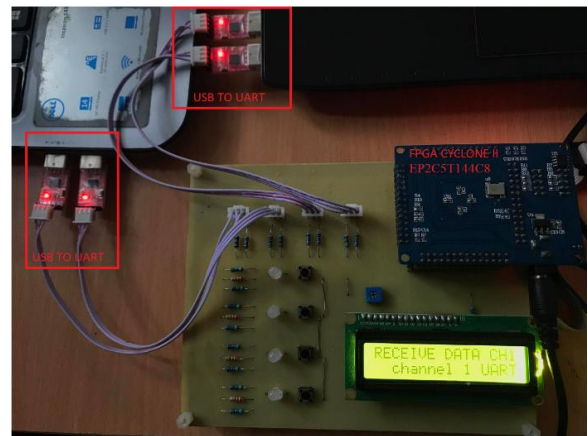


Hình 7. Giao tiếp bus APB.

Kết quả đánh giá qua mô phỏng và thực tế trên kit FPGA đã chỉ ra rằng module UART 4 kênh được thiết kế đảm bảo khả năng hoạt động ổn định, đáp ứng được các yêu cầu đề ra. Thiết kế đáp ứng tốt việc truyền nhận UART ở nhiều cấu hình khác nhau thể hiện qua các Testcase được thay đổi về tốc độ và cấu hình truyền.



Hình 8. Mạch kiểm tra thực tế.



Hình 9. Phần cứng kiểm tra thực tế.

4. KẾT LUẬN

Thiết kế được trình bày trong bài báo này sử dụng chuẩn bus AMBA 3 APB và là chuẩn bus đi kèm theo sự phát triển của dòng vi xử lý ARM. Mạch quản lý ngắt được thiết kế đơn giản và tiết kiệm tài nguyên hơn. Thiết kế mở rộng về cấu hình và cấu hình truyền hơn so với các bài báo đã có trước đây nhưng vẫn đảm bảo thiết kế tối ưu về tốc độ truyền ở mỗi kênh truyền. Thiết kế UART đa kênh được trình bày trong bài báo này có thể được sử dụng như một soft IP core ở quá trình Front End khi thiết kế IC để tạo Netlist và Function. Bên cạnh đó, trong khi hầu hết các thiết kế vi mạch ít được chia sẻ công khai, thiết kế UART đa kênh được trình bày trong bài báo này có giá trị cho việc học tập các môn liên quan tới thiết kế hệ thống số nói chung và thiết kế vi mạch nói riêng.

TÀI LIỆU THAM KHẢO

- [1] HU Zhe, ZHANG Jun, LOU Xi-ling, "A Novel Design of Efficient Multi-channel UART Controller Based on FPGA", Chinese Journal of Aeronautics, Volume 20, Issue 1, 2007, Pages 66-74.
- [2] G. B. Wakhle, I. Aggarwal and S. Gaba, "Synthesis and Implementation of UART Using VHDL Codes", International Symposium on Computer, Consumer and Control, Taichung, 2012, pp. 1-3.
- [3] Y. Fang and X. Chen, "Design and Simulation of UART Serial Communication Module Based on VHDL", 2011 3rd International Workshop on Intelligent Systems and Applications, Wuhan, 2011.
- [4] A. K. Gupta, A. Raman, N. Kumar and R. Ranjan, "Design and Implementation of High-Speed Universal Asynchronous Receiver and Transmitter (UART)", 7th International Conference on Signal Processing and Integrated Networks (SPIN), Noida, India, 2020, pp. 295-300.
- [5] Y. Wang and K. Song, "A new approach to realize UART", Proceedings of International Conference on Electronic & Mechanical Engineering and Information Technology, pp. 2749-2752, 2011.
- [6] U. Nanda and S. K. Pattnaik, "Universal Asynchronous Receiver and Transmitter (UART)", 3rd International Conference on Advanced Computing and Communication Systems (ICACCS), pp. 1-5, 2016.
- [7] M. Poorani and R. Kurunjimalar, "Design implementation of UART and SPI in single FPGA", 10th International Conference on Intelligent Systems and Control (ISCO), pp. 1-5, 2016.
- [8] AMBA™ Specification (Rev 2.0),
developer.arm.com/docs/ih0011/a/amba-specification-rev-20
- [9] AMBA™ 3 APB Protocol v1.0,
infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.ih0024b/index.html
- [10] Behrouz A. Forouzan and Sophia Chung Fegan, "Data Communication and Networking", 4th edition, McGraw Hill, 2007.
- [11] Stephen Brown and Zvonko Vranesic, "Fundamentals of Digital Logic with Verilog Design", third edition, McGraw Hill, 2014.
- [12] FPGA Cyclone II EP2C5T144C8 Datasheet,
datasheet.octopart.com/EP2C5T144C8-Altera-datasheet-7627975.pdf
- [13] G. Sung, L. Tung, H. Wang and J. Lin, "USB Transceiver With a Serial Interface Engine and FIFO Queue for Efficient FPGA-to-FPGA Communication", in IEEE Access, vol. 8, pp. 69788-69799, 2020.
- [14] Nguyen and X. Tran, "A novel asynchronous first-in-first-out adapting to multi-synchronous network-on-chips", International Conference on Advanced Technologies for Communications (ATC), 2014.
- [15] Pong P. Chu, "RTL Hardware Design using VHDL", A JOHN WILEY & SONS INC., 2006.
- [16] Realterm Serial Capture Program, <https://realterm.sourceforge.io/>

Tác giả chịu trách nhiệm bài viết:

TS. Đỗ Duy Tân

Trường ĐH Sư Phạm Kỹ Thuật TPHCM

Email: tandd@hcmute.edu.vn