|  |
| --- |
| TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI  **VIỆN ĐIỆN TỬ - VIỄN THÔNG**  logo_128  BÁO CÁO BÀI TẬP LỚN  **HỆ THỐNG NHÚNG**  **Đề tài: Thiết kế hệ thống điều chế truyền thông FSK Theo chuẩn uart sử dụng transducer trên FPGA**  Nhóm sinh viên thực hiện: Nhóm 11  Đàm Thanh Sơn MSSV: 20172794  Lê Đắc Khánh MSSV: 20172621  Nguyễn Sĩ Linh MSSV: 20172654  Vũ Văn Đại MSSV: 20172445  Phạm Văn Chiến MSSV: 20172434  **Giảng viên hướng dẫn: TS. Hàn Huy Dũng**  Hà Nội, 6-2021 |

# Lời mở đầu

Trên thế giới hiện nay, truyền thông dưới nước đang ngày càng phát triển được nghiêm cứu sâu hơn để áp dụng cho rất nhiều mục đích khác nhau trong thực tế, như trong thăm dò tài nguyên biển, định vị dẫn đường trên biển, cho liên lạc quân sự trên biển… Trên thế giới là như vậy còn tại Việt Nam chúng ta với đường bờ biển trải dài hàng nghìn ki lô mét, dù mục đích có là thăm dò tài nguyên biển, định vị dẫn đường trên biển hay mục đích quân sự cũng đều đáng được quan tâm.

Khi nghiêm cứu, xem xét về các mô hình truyền thông tin dưới nước, tuy có rất nhiều đặc tính, thông số môi trường đặc thù, ảnh hưởng lên kênh truyền khác hẳn với môi trường không gian tự do. Thế nhưng từ những kiến thức và thông tin về vô tuyến đồ sộ và dựa vào đó ta có thể đưa ra những phương pháp mô hình truyền kênh có thể phục vụ tốt cho lĩnh vực này. Nhằm phục vụ cho việc phát triển và nghiêm cứu, cũng như phục vụ cho môn học nhóm em xin giới thiệu đề tài “Thiết kế hệ thống phát thông tin dưới nước sử dụng transducer trên FPGA” làm đề tài thực hiện bài tập lớn của nhóm.

Trong quá trình làm bài tập lớn, nhóm em đã cố gắng để hoàn thiện đề tài một cách tốt nhất. Nhưng với kiến thức và sự am hiểu còn hạn chế nên chắc sẽ không tránh khỏi những thiếu sót. Em mong thầy và các bạn có thể đóng ghóp thêm ý kiến để đề tài của bọn em được hoàn thiện hơn.

Nhóm em xin gửi lời cảm ơn tới TS. Hàn Huy Dũng đã nhiệt tình hướng dẫn, giúp đỡ, tạo điều kiện tốt nhất về kiến thức cũng như tinh thần để chúng em hoàn thành tốt bài tập lớn này.

Nhóm thực hiện

Nhóm 11

MỤC LỤC

[Lời mở đầu 2](#_Toc81853428)

[CHƯƠNG 1. TỔNG QUAN VỀ FPGA 7](#_Toc81853429)

[1.1 Khái niệm FPGA 7](#_Toc81853430)

[1.2 Cấu trúc cơ bản của FPGA 8](#_Toc81853431)

[1.3 Khối logic 8](#_Toc81853433)

[1.3.1 Hệ thống mạch liên kết 9](#_Toc81853436)

[1.3.2 Các phần tử tích hợp sẵn 10](#_Toc81853438)

[1.3.3 . Block RAM 10](#_Toc81853439)

[1.4 Quy trình thiết kế FPGA 10](#_Toc81853440)

[1.4.1 Mô tả thiết kế (Specitification) 10](#_Toc81853441)

[1.4.2 Thiết kế (Design) 10](#_Toc81853442)

[1.4.3 Mô phỏng logic (Function simulation) 11](#_Toc81853443)

[1.4.4 Tổng hợp mạch logic (Logic Synthesis) 11](#_Toc81853444)

[1.4.5 Hiệu chỉnh các kết nối (Datapath Schematic) 11](#_Toc81853445)

[1.4.6 Thực thi (Implementation) 11](#_Toc81853446)

[1.4.7 Quá trình lập trình và nạp chương trình 12](#_Toc81853447)

[1.5 Giới thiệu Kit Xilinx Spartan 6 Nexys 3 12](#_Toc81853448)

[CHƯƠNG 2. Chương 2. Lý thuyết về FSK 14](#_Toc81853450)

[2.1 Khái niệm 14](#_Toc81853451)

[2.2 Đặc điểm 14](#_Toc81853452)

[2.3 Đánh giá và ứng dụng 15](#_Toc81853456)

[2.3.1 Đánh giá 15](#_Toc81853457)

[2.3.2 Ứng dụng 15](#_Toc81853458)

[CHƯƠNG 3. THIẾT KẾ 17](#_Toc81853459)

[3.1 Xác định yêu cầu 17](#_Toc81853460)

[3.2 Tổng quan hệ thống 17](#_Toc81853461)

[3.2.1 Các chỉ tiêu kỹ thuật: 18](#_Toc81853463)

[3.2.2 Yêu cầu hệ thống 18](#_Toc81853464)

[3.3 Lập kế hoạch 19](#_Toc81853466)

[3.4 Thiết kế hệ thống 21](#_Toc81853468)

[3.4.1 Mô hình hệ thống 21](#_Toc81853469)

[3.5 Thiết kế chi tiết 22](#_Toc81853472)

[3.5.1 Khối chuẩn giao tiếp UART 22](#_Toc81853473)

[3.5.2 FSK\_mod 27](#_Toc81853482)

[CHƯƠNG 4. LINH KIỆN VÀ VI ĐIỀU KHIỂN 32](#_Toc81853491)

[4.1 Lựa chọn linh kiện 32](#_Toc81853492)

[4.2 Thông số và chức năng linh kiện 32](#_Toc81853493)

[4.2.1 Khối FSK\_Mod 32](#_Toc81853494)

[**Hình 4.1 Kit Nexys Xilinx Spartan 6 Nexys 3** 33](#_Toc81853496)

[4.2.2 Khối khuếch đại 34](#_Toc81853497)

[4.2.3 Khối Tranducer 35](#_Toc81853500)

[4.2.4 Giá thành linh kiện. 35](#_Toc81853501)

[CHƯƠNG 5. Kết quả 37](#_Toc81853503)

[CHƯƠNG 6. KẾT LUẬN 38](#_Toc81853505)

[CHƯƠNG 7. Tài liệu tham khảo 39](#_Toc81853506)

**DANH MỤC HÌNH**

[Hình 1. 1 Cấu trúc cơ bản FPGA 5](#_Toc81853306)

[Hình 1. 2 Khối logic 5](#_Toc81853308)

[Hình 1. 3 Configurable Logic Blocks 6](#_Toc81853309)

[Hình 1. 4 Mạng liên kết trong FPGA 6](#_Toc81853311)

[Hình 1. 5 Kit Nexys 3 9](#_Toc81853323)

[Hình 2. 1 FSK modulation 11](#_Toc81853327)

[Hình 2. 2 Phổ của tín hiệu FSK 12](#_Toc81853328)

[Hình 2. 3 Phương pháp điều chế 12](#_Toc81853329)

[Hình 3. 1 Ý tưởng thiết kế FSK\_mod 14](#_Toc81853336)

[Hình 3.2 Sơ đồ hệ thống phát tín hiệu dưới nước 21](#_Toc81853470)

[Hình 3. 3 Sơ đồ chân khối FSK\_mod 22](#_Toc81853471)

[Hình 3.4 Sơ đồ UART 23](#_Toc81853474)

[Hình 3.5 Sơ đồ bit truyền 23](#_Toc81853475)

[Hình 3.6 Khối baud gate 24](#_Toc81853476)

[Hình 3.7 Khối thu RX 24](#_Toc81853477)

[Hình 3.8 Sơ đồ thuật toán khối RX 25](#_Toc81853478)

[Hình 3.9 Khối thu TX 26](#_Toc81853479)

[Hình 3.10 Sơ đồ thuật toán khối TX 27](#_Toc81853480)

[Hình 3.11 Khối FIFO 27](#_Toc81853481)

[Hình 3.12 Khối FSK\_Mod 27](#_Toc81853483)

[Hình 3. 13 Sơ đồ chân khối Bit\_rate\_1Khz 28](#_Toc81853484)

[Hình 3.14 Sơ đồ chân khối sin\_data 28](#_Toc81853486)

[Hình 3. 15 SPI timing MCP 4921 29](#_Toc81853488)

[Hình 3. 16 Sơ đồ chân khối dac\_out 29](#_Toc81853489)

[Hình 4.2 Transistor BC547 34](#_Toc81853498)

[Hình 4.3 Op Amp LF356N 35](#_Toc81853499)

[Hình 5. 1 Wareform 37](#_Toc81853504)

**DANH MỤC BẢNG**

[Bảng 3.1. Bảng tính NRE Cost 18](#_Toc81853465)

[Bảng 3.2 Bảng phân công công việc 19](#_Toc81853467)

[Bảng 4.1 Thông số của các kit trên thị trường 32](#_Toc81853495)

[Bảng 4.2 Bảng giá thành linh kiện 35](#_Toc81853502)

# TỔNG QUAN VỀ FPGA

## Khái niệm FPGA

Field-programmable gate array (FPGA) là một loại mạch tích hợp cỡ lớn dùng cấu trúc mảng phần tử logic mà người dùng có thể lập trình được. Chữ “field” ở đây muốn chỉ đến khả năng tái lập trình từ bên ngoài của người sử dụng, không phụ thuộc vào dây truyền sản xuất phức tạp của nhà máy bán dẫn. Vi mạch FPGA được cấu thành từ các bộ phận như:

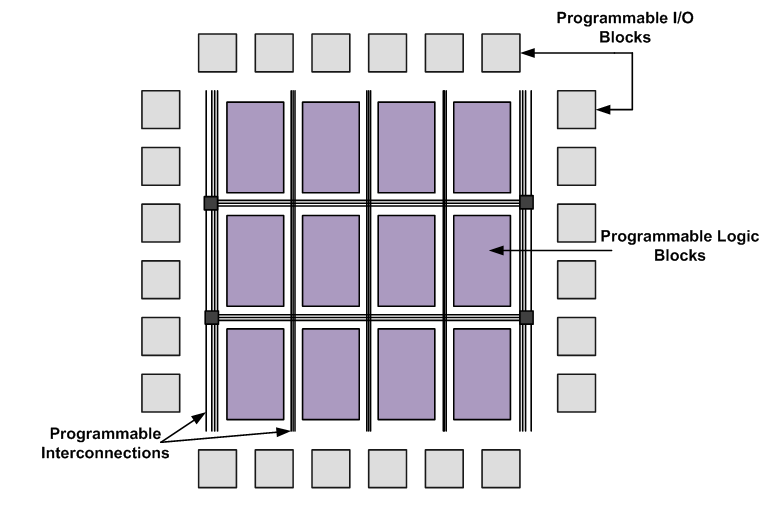
* Các khối logic cơ bản lập trình được (logic block)
* Hệ thống mạch liên kết lập trình được
* Khối vào/ ra (I/O pads)
* Phần tử thiết kế sẵn như DSP slice, RAM, ROM, nhân xử lý, …

FPGA cũng được xem như một loại vi mạch bán dẫn chuyên dụng ASIC, nhưng nếu so sánh FPGA với những ASIC đặc chế hoàn toàn hay ASIC thiết kế trên thư viện logic thì FPGA không đạt được mức độ tối ưu như những loại này và hạn chế trong khả năng thực hiện những tác vụ đặc biệt phức tạp, tuy vậy FPGA ưu việt hơn ở chỗ có thể tái cấu trúc lại khi đang sử dụng, công đoạn thiết kế đơn giản, do vậy chi phí giảm, rút ngắn thời gian đưa sản phẩm vào sử dụng.

Còn nếu so sánh với các dạng vi mạch bán dẫn lập trình được dùng cấu trúc mảng phần tử logic như PLA, PAL, CPLD thì FPGA ưu việt hơn ở các điểm: tác vụ lập trình của FPGA thực hiện đơn giản hơn, khả năng lập trình linh động hơn và khác biệt quan trọng nhất là kiến trúc của FPGA cho phép nó có khả năng chứa khối lượng lớn các cổng logic (logic gate), so với các vi mạch bán dẫn lập trình được có trước đó.

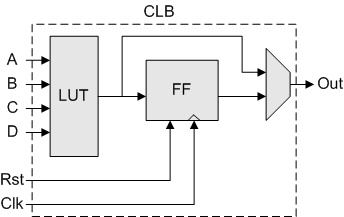
Thiết kế hay lập trình cho FPGA được thực hiện chủ yếu bằng các ngôn ngữ mô tả phần cứng HDL như VHDL, Verilog, …. Các hang sản suất FPGA lớn như Xilinx, Altera thường cung cấp các gói phần mềm và thiết bị phụ trợ cho quá trình thiết kế, ngoài ra cũng có một số hang thứ 3 cung cấp các gói phần mềm kiểu này như Synopsys, Synplify,… Các gói phần mềm này có khả năng thực hiện tất cả các bước của toàn bộ quy trình thiết kế IC chuẩn với đầu vào là mã thiết kế trên HDL (còn gọi là mã RTL).

## Cấu trúc cơ bản của FPGA



Hình 1. 1 Cấu trúc cơ bản FPGA

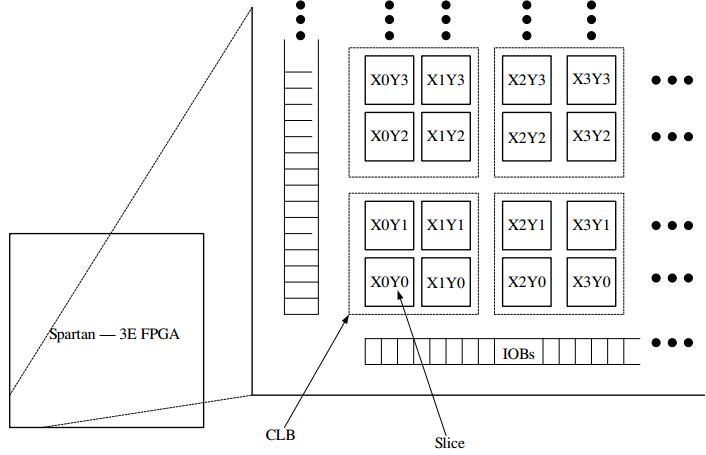
## Khối logic



Hình 1. 2 Khối logic

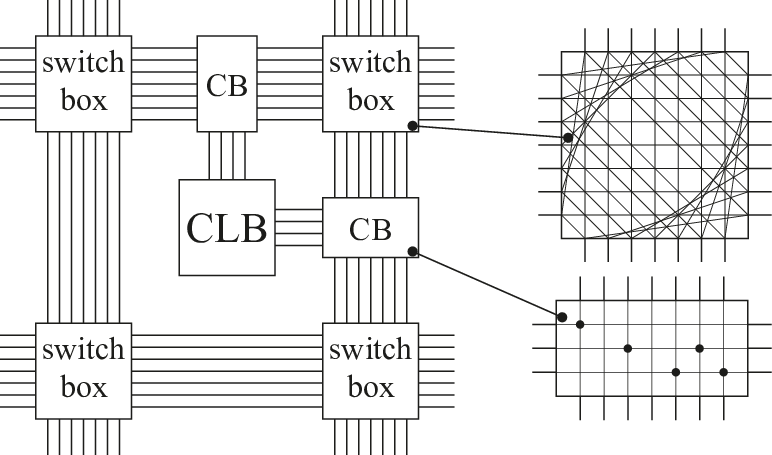
Phần tử chính của FPGA là các khối logic (logic block). Khối logic được cấu thành từ LUT và một phần tử nhớ đồng bộ Flip-flop, LUT (Look up table) là khối logic có thể thực hiện bất kỳ hàm logic nào từ 4 đầu vào, kết quả của hàm này tùy vào mục đích mà gửi ra ngoài khối logic trực tiếp hay không thông qua phần tử nhớ Flip-flop.

Các dòng FPGA của Xilinx còn sử dụng khái niệm slice, 1 slice tạo thành từ 2 khối logic, số lượng slice thay đổi từ vài nghìn tới vài chục nghìn tùy theo loại FPGA. Bốn slice tạo thành một khối Configurable Logic Blocks (CLBs). CLBs là phần tử cơ bản cấu thành FPGA, là phần tử cơ bản cấu thành FPGA, là nguồn tài nguyên logic chính tạo nên các mạch logic. Nếu nhìn cấu trúc tổng thể của mảng LUT thì ngoài 4 đầu vào kể trên còn hỗ trợ them 2 đầu vào bổ sung từ các khối logic phân bố trước và sau nó nâng tổng số đầu vào của LUT lên 6 chân. Cấu trúc này nhằm tang tốc các bộ số học logic.



Hình 1. 3 Configurable Logic Blocks

### Hệ thống mạch liên kết



Hình 1. 4 Mạng liên kết trong FPGA

Mạng liên kết trong FPGA được cấu thành từ các đường kết nối theo hai phương ngang cà đứng, tùy theo từng loại FPGA mà các đường kết nối được chia thành các nhóm khác nhau, ví dụ trong XC4000 của Xilinx có ba loại kết nối: ngắn, dài và rất dài. Các đường kết nối được nối với nhau thông qua các khối chuyển mạch lập trình được (programmable switch), trong một khối chuyển mạch chứa một số lượng nút chuyển lập trình được đảm bảo cho các dạng liên kết phức tạp khác nhau.

### Các phần tử tích hợp sẵn

Ngoài các khối logic tùy theo các loại FPGA khác nhau mà có các phần tử tích hợp thêm khác nhau, ví dụ để thiết kế những ứng dụng SoC, trong dòng Virtex 4, 5 của Xilinx có chứa nhân Power PC, hay trong dòng Atmel FPSLIC tích hợp nhân AVR,… hay trong những ứng dụng xử lý tín hiệu số trong FPGA có tích hợp sẵn các DSP slices là bộ nhân cộng tốc độ cao, thực hiện hàm A \* B + C, ví dụ dòng Virtex của Xilinx chứa từ vài chục đến hàng tram DSP slices với A, B, C 18-bit.

### . Block RAM

Ngoài ra những FPGA của Xilinx còn có block RAM, có thể khái quát Block RAM là bộ nhớ nhỏ nằm trong FPGA. Những block này tuy nhỏ (khoảng vài chục Kb đến vài Mb tùy theo loại FPGA) nhưng có thể tạo thành bộ nhớ nhỏ như ROM, FIFO.

## Quy trình thiết kế FPGA

### Mô tả thiết kế (Specitification)

Khi xây dựng một chip khả trình (FPGA) với ý nghĩa dành cho một ứng dụng riêng biệt, vì xuất phát từ mỗi ứng dụng trong thực tiễn cuộc sống sẽ đặt ra yêu cầu phải thiết kế IC thực hiện tối ưu nhất những ứng dụng đó. Bước đầu tiên của quy trình thiết kế này có nhiệm vụ tiếp nhận các yêu cầu của thiết kế và xây dựng nên kiến trúc tổng quát của thiết kế.

### Thiết kế (Design)

Trong bước này, từ những yêu cầu của thiết kế và dựa trên khả năng của công nghệ hiện có, người thiết kế kiến trúc sẽ xây dựng nên toàn bộ kiến trúc tổng quan cho thiết kế. Nghĩa là trong phần này người thiết kế kiến trúc phải mô tả được những vấn đề sau:

* Thiết kế có những khối nào?
* Mỗi khối có chức năng gì?
* Hoạt động của mỗi khối ra sao?
* Phân tích các kỹ thuật sử dụng trong thiết kế và các công cụ, phần mềm bổ trợ cho việc thiết kế.

Một thiết kế có thể được mô tả sử dụng ngôn ngữ mô tả phần cứng như VHDL hay Verilog hoặc có thể mô tả qua bảng vẽ mạch để mô tả chi tiết cho các khối trong sơ đồ.

### Mô phỏng logic (Function simulation)

Sau khi mô tả thiết kế, người thiết kế cần mô phỏng tổng thể thiết kế về mặt chức năng để kiểm tra thiết kế có hoạt động đúng với các chức năng yêu cầu.

### Tổng hợp mạch logic (Logic Synthesis)

Tổng hợp là quá trình tổng hợp các mô tả thiết kế thành sơ đồ bố trí mạch (netlist). Quá trình này chia làm 2 bước:

Chuyển đổi các mã RTL, mã HDL thành mô tả dưới dạng các biểu thức đại số Boolean.

Dựa trên các biểu thức này kết hợp với thư viện tế bào chuẩn hóa sẵn để tổng hợp nên một thiết kế tối ưu.

### Hiệu chỉnh các kết nối (Datapath Schematic)

Nhập netlist và các ràng buộc về thời gian vào một công cụ phân tích thời gian (timing analysic), công cụ này sẽ tách rời tất cả các kết nối của thiết kế, tính thời gian trễ của các kết nối dựa trên các ràng buộc. Dựa trên kết quả phân tích (report) của công cụ phân tích, xác định các kết nối không thỏa mãn về mặt thời gian. Tùy theo nguyên nhân dẫn đến không thỏa mã mà ta có thể viết lại mã và tiến hành lại tổng hợp mạch logic hoặc hiệu chỉnh lại các ràng buộc.

### Thực thi (Implementation)

Quá trình gồm các bước:

* Ánh xạ (mapping): chuẩn bị dữ liệu đầu vào, xác định kích thước của các khối. Các khối này sẽ phải phù hợp với cấu trúc của 1 tế bào cơ bản FPGA (gồm nhiều cổng logic) và đặt chúng vào các vị trí tối ưu cho việc đi dây.
* Sắp đặt và định tuyến (Place and route): Sắp đặt các khối là đặt các khối đó vào các tế bào ở vị trí tối ưu cho việc đi dây. Định tuyến là bước thực hiện việc nối dây các tế bào.

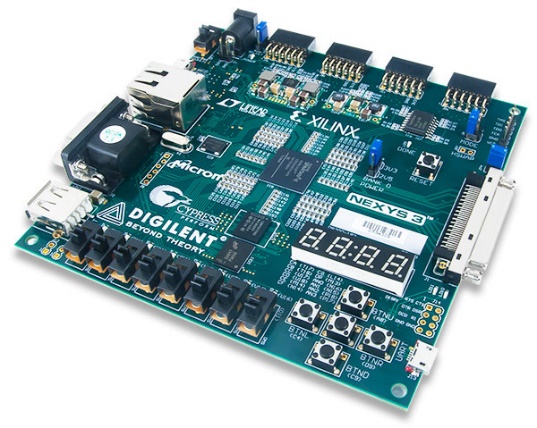
Để thực hiện các công việc này chúng ta cần có các thông tin như sau:

* Các thông tin vật lý về thư viện tế bào, ví dụ kích thước tế bào, các điểm để kết nối, định thời, các trở ngại trong khi đi dây.
* Một netlist được tổng hợp sẽ chỉ ra chi tiết các instance và mối quan hệ kết nối bao gồm cả các đường dẫn bị hạn chế trong thiết kế,
* Tất cả các yêu cầu của tiến trình cho các lớp kết nối, bao gồm các luật thiết kế cho các lớp đi dây, trở kháng và điện dung, tiêu thụ năng lượng, các luật về sự dẫn điện trong mỗi lớp.

### Quá trình lập trình và nạp chương trình

Sau quá trình thực hiện thiết kế, thiết kế đó cần được được nạp vào FPGA dưới dạng bit. Quá trình nạp thiết kế vào FPGA sẽ được lưu trữ trong bộ nhớ tạm, và sẽ mất khi tắt nguồn. Ở đây để thực hiện triển khai các thiết kế trên KIT FPGA Nexys 3 thì chúng ta sử dụng phần mềm ISE của Xilinx để tạo file nạp và dùng phần mềm Digilent Adept để nạp file bit xuống kit FPGA Nexys 3.

## Giới thiệu Kit Xilinx Spartan 6 Nexys 3



Hình 1. 5 Kit Nexys 3

Các thông số của kit:

* Kit hoạt động ở tần số 100MHz
* 16MB Cellular RAM
* 16MB SPI
* 10/100 Ethernet PHY
* USB2 cho chạy chương trình và truyền dữ liệu xfer
* USB-UART và USB-HID
* 8 bit VGA
* GPIO: 8 LEDs, 5 nút bấm, 8 công tắc và 4 led 7 thanh
* 2278 slices, mỗi slices bao gồm 4 LUT 6 đầu vào và 8 Flip-flop
* 576Kb cho khối RAM phục vụ xử lý nhanh
* 32 DSP

Đánh giá:

Kit phục vụ tốt cho nhu cầu của sinh viên và phòng LAB. Tần số hoạt động cao hơn kit DE2 giúp cho việc xử lý trong các khối hoạt động nhanh hơn.

# Chương 2. Lý thuyết về FSK

## Khái niệm

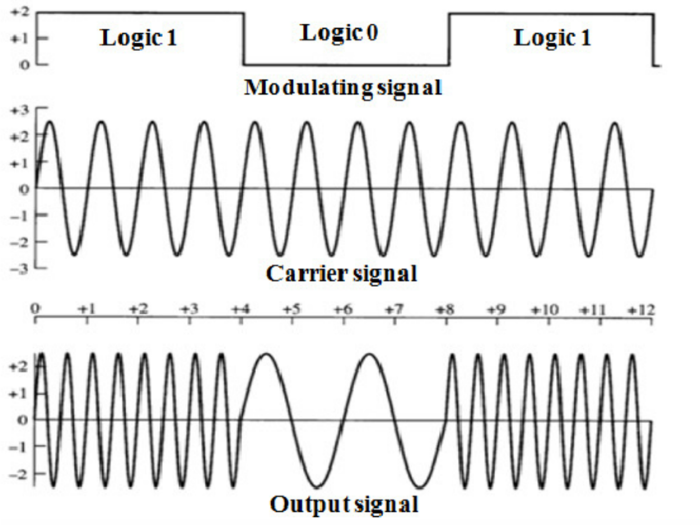
Frequency Shift Keying (FSK) là sơ đồ điều chế tần số trong đó thông tin số được truyền qua các thay đổi tần số riêng biệt của tín hiệu sóng mang.

Công nghệ này được sử dụng cho các hệ thống thông tin liên lạc như đo từ xa, radio , user ID, dụng cụ mở cửa nhà để xe và truyền vô tuyến tần số thấp trong các băng tần VLF và ELF.

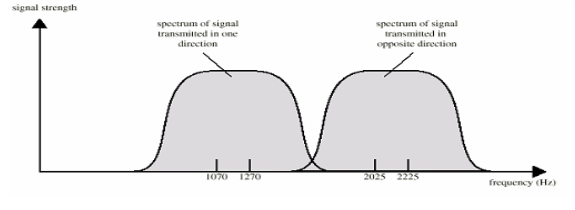
FSK đơn giản nhất là FSK nhị phân (BFSK). BFSK sử dụng một cặp tần số riêng biệt để truyền thông tin nhị phân (0 và 1).

## Đặc điểm

* Dùng nhiều hơn hai tần số
* Băng thông được dùng hiệu quả hơn
* Hay mắc lỗi khi truyền
* Mỗi phần tử tín hiệu nhiều hơn 1 bit dữ liệu.



Hình 2. 1 FSK modulation



Hình 2. 2 Phổ của tín hiệu FSK

A picture containing object, clock

Description automatically generated

Hình 2. 3 Phương pháp điều chế

## Đánh giá và ứng dụng

### Đánh giá

Ưu điểm:

* Ít bị ảnh hưởng bởi nhiễu và ít lỗi hơn so với ASK

Nhược điểm:

* Tần số cao dễ bị nhiễu và hạn chế tốc độ truyền
* Khó đồng bộ

### Ứng dụng

Kỹ thuật FSK có các ứng dụng trong thông tin vô tuyến như sau:

* Dùng rộng rãi trong truyền số liệu
* Dùng để truyền dữ liệu tốc độ 1200bps hay thấp hơn trên mạng điện thoại
* Có thể dùng tần số cao (3-30MHz) để truyền sóng radio hoặc cáp đồng trục

# THIẾT KẾ

## Xác định yêu cầu

Xác định nhu cầu thị trường:

* Việc thế kế vô tuyến bằng cấu hình phần mềm đang phát triển mạnh
* Khả năng thích ứng, tái sử dụng cao, cấu hình lại theo yêu cầu
* Điều chế tín hiệu trước khi truyền đi xa trong môi trường nước

Đối tượng:

* Tàu ngầm, tàu thuyền có nhu cầu truyền và giao tiếp thông tin trong môi trường nước.

Lý do chọn đề tài:

* Củng cố kiến thức môn học hệ thống nhúng và thiết kế giao tiếp nhúng, nâng cao kỹ năng lập trình và thiết kế hệ thống nhúng.
* Tương tai nhóm chúng em muốn hoàn thiện hoàn chỉnh hệ thống truyền thông dưới nước để đáp ứng với nhu cầu của thực tế, tạo ra cơ hội phát triển các ý tưởng và sản phẩm bắt kịp với xu hướng mới, nhưng do điều kiện về vật chất cũng như thời gian có hạn nên nhóm mới chỉ dừng lại ở phạm vi đề tài.

## A screenshot of a video game Description automatically generatedTổng quan hệ thống

Hình 3. 1 Ý tưởng thiết kế FSK\_mod

### Các chỉ tiêu kỹ thuật:

* Tốc độ bit: 1 Kbps
* Tần số sóng mang của bit 1: 16 Khz
* Tần số sóng mang của bit 0: 13 Khz

### Yêu cầu hệ thống

#### Yêu cầu chức năng

* Dữ liệu cần truyền đi là 8 bit
* Tốc độ bit: 1Kbps
* Tần số sóng mang của bit 1: 16 Khz
* Tần số sóng mang của bit 0: 13 Khz

#### Yêu cầu phí chức năng

* Khoảng cách truyền <10km
* Giá thành <100$
* Vỏ bọc: nhựa chất liệu chống ăn mòn
* Độ trễ truyền: < 5s
* Khối lượng: 1.5 kg
* Công suất: 10W
* Độ bền: 5 năm
* Nhiệt độ môi trường: 10-40
* Chuẩn chống nước: IP68
* Size: 30x20x10 (cm)
* NRE Cost: 10000$
* Time to market: 3 tháng
* Time to prototype: 2 tháng

***Bảng 3.1. Bảng tính NRE Cost***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Danh mục chi | Số tiền 1 tháng ($) | Số lượng | Số tiền 3 tháng |
| 1 | Thuế văn phòng | 300 | 1 | 900 |
| 2 |  | 700 | 1 Team Leader | 2100 |
|  | Thanh toán lương cho KS | 500 | 4 Fresher | 6000 |
| 3 | Trang thiết bị nghiêm cứu |  | Máy đo, linh kiên | 1000 |
| 4 |  |  | Total | 10000 |

## Lập kế hoạch

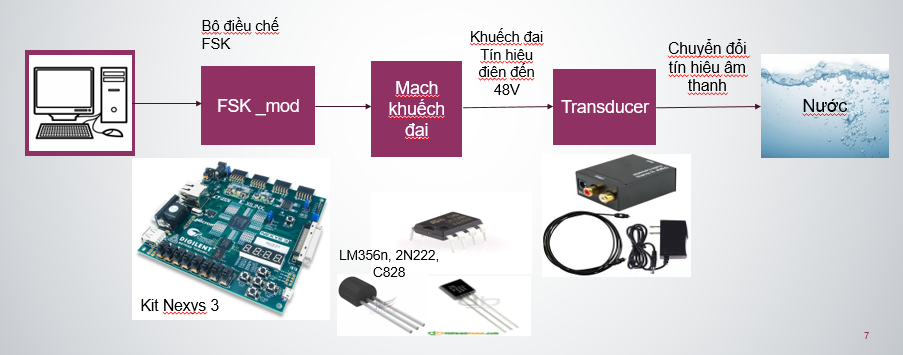
Để có thể hoàn thành bài tập lớn một cách hiệu quả nhất với thời gian nhanh nhất, chúng em đã phân tích điểm mạnh và điểm yếu của từng thành viên trong nhóm, căn cứ vào đó phân chia công việc cụ thể phù hợp với từng thành viên được thể hiện trong dưới đây:

### Bảng 3.2 Bảng phân công công việc

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên công việc** | **Thời gian**  **thực hiện** | **Người phụ trách** | **Trạng thái** |
| Tìm kiếm đề tài thực tế, phân tích nhu cầu thị trường, tham khảo Project | 1 ngày  (22/3 -23/3) | Đàm Thanh Sơn  Phạm Văn Chiến | Xong |
| So sánh, phân tích, đưa ra quyết định chọn đề tài: Điều chế FSK trên FPGA | 1 ngày  (23/3 - 23/3) | Tất cả các thành viên | Xong |
| Tìm hiểu lý thuyết về FSK   * Frequency Shift Keying (FSK) * Tham khảo từng cách sử dụng cho các hệ thống thông tin liên lạc đi xa, radio , user ID, dụng cụ mở cửa nhà để xe và truyền vô tuyến tần số thấp. * Tìm hiểu FSK nhị phân (BFSK). BFSK truyền thông tin nhị phân (0 và 1). | 3 ngày  (23/3 - 26/3) | Nguyễn Sĩ Linh  Lê Đắc Khánh  Phạm Văn Chiến | Xong |
| Đánh giá và so sánh với ASK   * Về ưu điểm:Tại sao lại ít ảnh hưởng của nhiễu và ít lỗi hơn ASK. * Về nhược điểm: Tại sao lại khó đồng bộ và hạn chế tốc độ truyền ở tần số cao. | 3 ngày  (23/3 - 26/3) | Nguyễn Sỹ Linh  Vũ Văn Đại | Xong |
| Tìm hiểu về ứng dụng của FSK | 3 ngày  (23/3 - 26/3) | Vũ Văn Đại  Phạm Văn Chiến | Xong |
| Tìm hiểu tổng quan hệ thống:  Các chỉ tiêu kỹ thuật | 3 ngày  (23/3-26/3) | Đàm Thanh Sơn | Xong |
| Thiết kế hệ thống | 1 tuần  (26/3 - 2/4) | Đàm Thanh Sơn  Phạm Văn Chiến | Xong |
| Thiết kế chuẩn giao tiếp UART | 1 tuần  (2/4 – 9/4) | Đàm Thanh Sơn | Xong |
| Thiết kế khối Bit\_rate\_1khz | 1 tuần  (9/4 - 16/4) | Đàm Thanh Sơn  Vũ Văn Đại | Xong |
| Thiết kế khối sin\_data | 1 tuần  (16/4 - 23/4) | Đàm Thanh Sơn  Nguyễn Sỹ Linh | Xong |
| Thiết kế khối dac\_out | 1 tuần  (23/4 - 30/4) | Đàm Thanh Sơn  Phạm Văn Chiến | Xong |
| Mô phỏng trên ModelSim | 1 tuần  (30/4 - 7/5) | Đàm Thanh Sơn | Xong |
| Triển khai trên FPGA | 2 tuần  (7/5 - 21/5) | Đàm Thanh Sơn | Chưa h/t do dịch Covid-19 |
| Hoàn thiện báo cáo slide và word | 20 ngày  (21/5 - 10/6) | Đàm Thanh Sơn  Lê Đắc Khánh  Phạm Văn Chiến | Xong |

## Thiết kế hệ thống

### Mô hình hệ thống

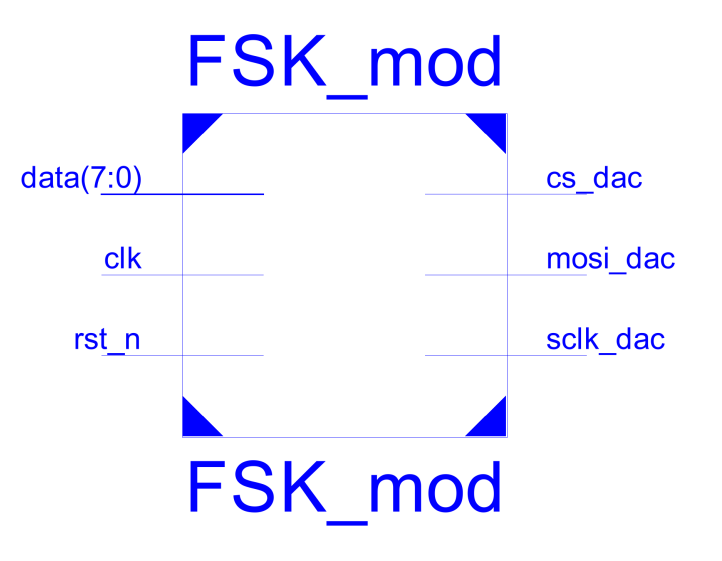


**Hình 3.2 Sơ đồ hệ thống phát tín hiệu dưới nước**

Hệ thống truyền thông tin dưới bao gồm:

* Bộ điều chế FSK ( Kit Nexys 3)
* Mạch khuếch đại ( Khuếch đại tín hiệu điện)
* Transducer (chuyển đổi tín hiệu điện thành tín hiệu âm thanh)

Từ lý thuyết FSK đã trình bày ở trên, em đưa ra thiết kế như sau:



Hình 3. 3 Sơ đồ chân khối FSK\_mod

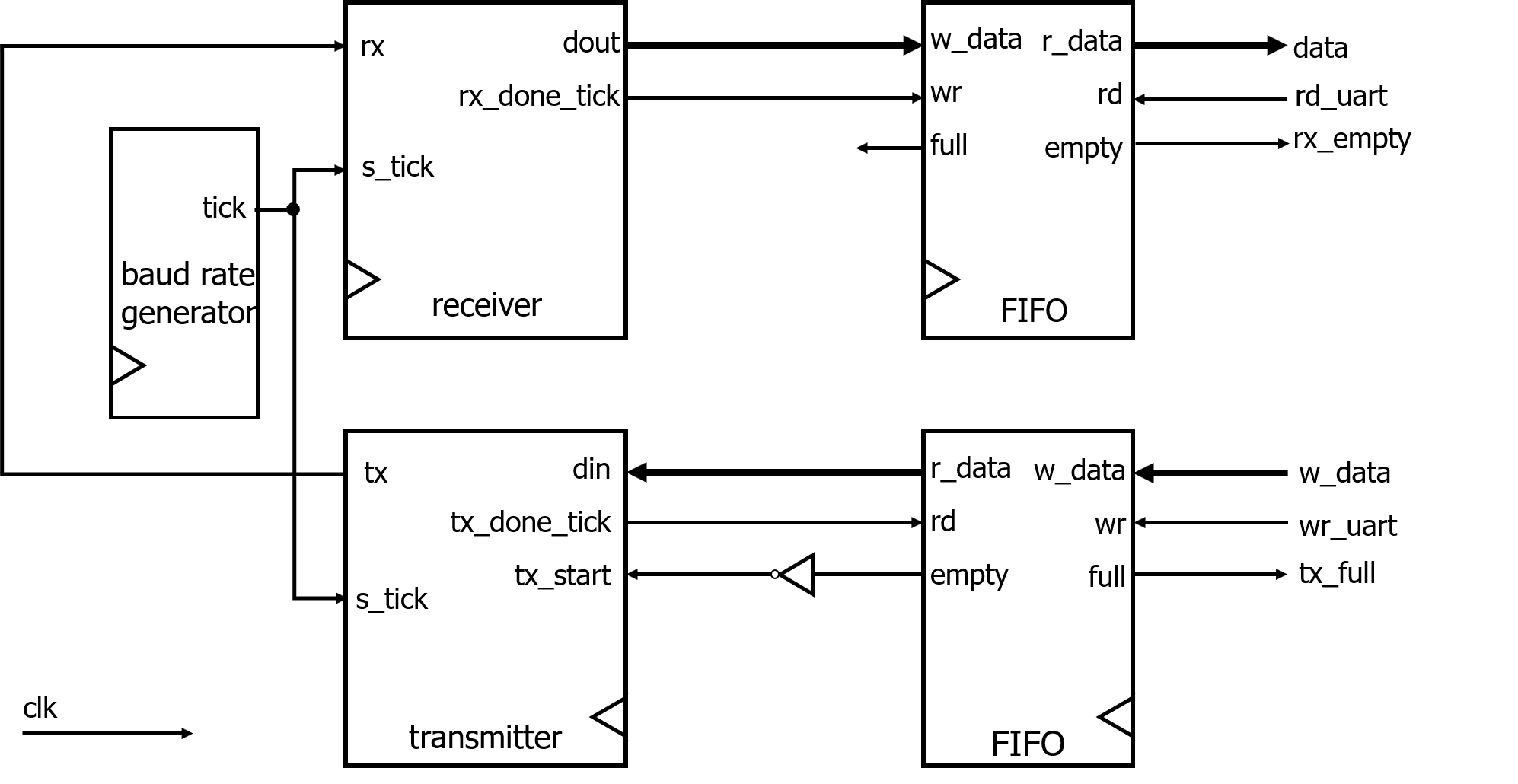
Chức năng của các port là:

* Data (7:0): nhận giá trị tín hiệu cần truyền đi
* Clk: điều khiển hoạt động của mạch
* Rst\_n: reset mạch
* Cs\_dac, mosi\_dac, sclk\_dac: là 3 chân tín hiệu giao tiếp SPI để truyền dữ liệu sang DAC để chuyển đổi tín hiệu số sang tín hiệu tương tự

## Thiết kế chi tiết

### Khối chuẩn giao tiếp UART

Trước khi đưa được tín hiệu vào khối FSK\_mod chúng ta cần chuẩn giao tiếp với máy tính, ở đây em sử dụng chuẩn giao tiếp UART.

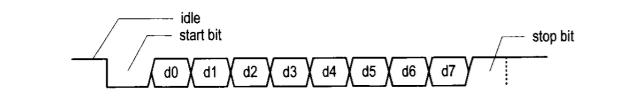


### Hình 3.4 Sơ đồ UART

UART gồm 4 khối chính được nối như hình

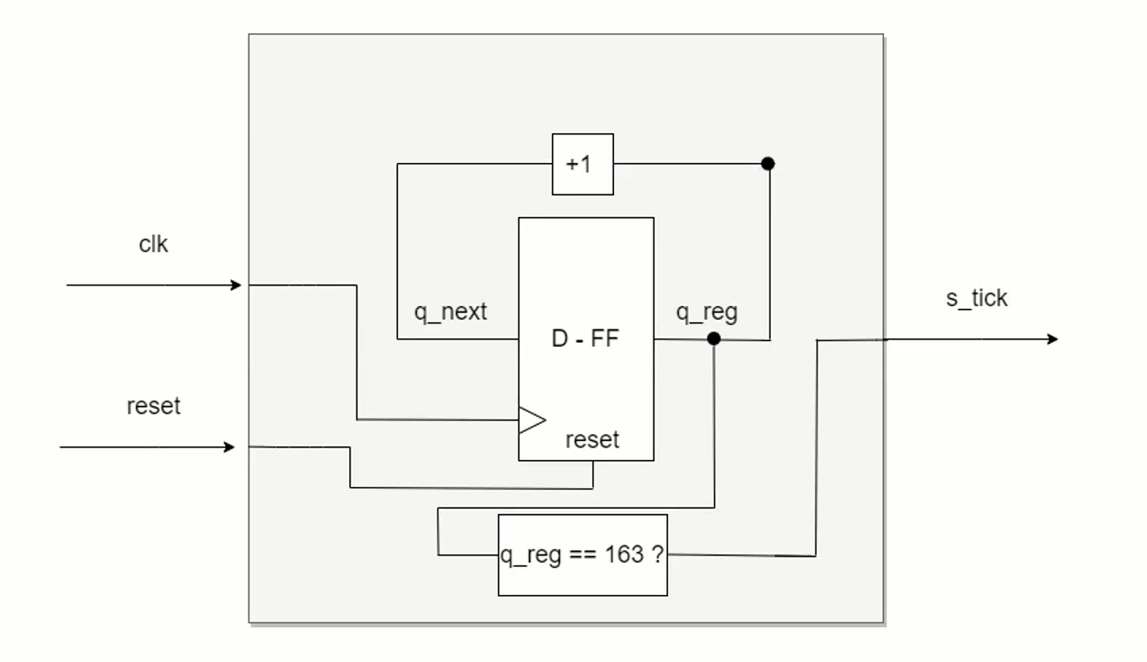
Thuật toán:

1. Tín hiệu ban đầu luôn ở mức cao, chờ cho đến khi tín hiệu chuyển xuống mức 0 chính là điểm bắt đầu của bit start, bộ đếm tick bắt đầu hoạt động, thực hiện đếm lấy mẫu tín hiệu.
2. Khi bộ đếm tick đạt tới 7, lúc này đang ở chính giữa tín hiệu start, thực hiện nhận bit start và khởi động bộ đếm dữ liệu.
3. Khi bộ đếm dữ liệu đạt tới giá trị 15, lúc này đang ở chính giữa bit dữ liệu thứ nhất, nhận bit dữ liệu và chuyển nó vào thanh ghi đồng thời reset lại bộ đếm,
4. Lặp lại bước 3 thêm 7 lần để nhận toàn bộ 8 bit dữ liệu.
5. Nếu sử dụng bit parity thì lặp bước 3 thêm 1 lần nữa để nhận bit parity.
6. Lặp lại bước 3 một lần nữa để nhận nốt bit stop.



**Hình 3.5 Sơ đồ bit truyền**

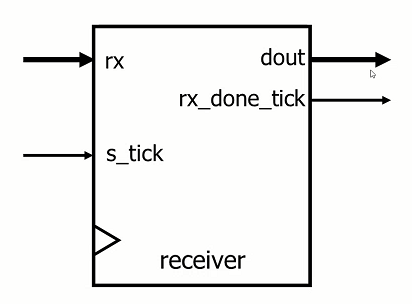
#### Khối Baud gate



### Hình 3.6 Khối baud gate

Khối baud gate có nhiệm vụ đồng bộ khối RX và khối TX.

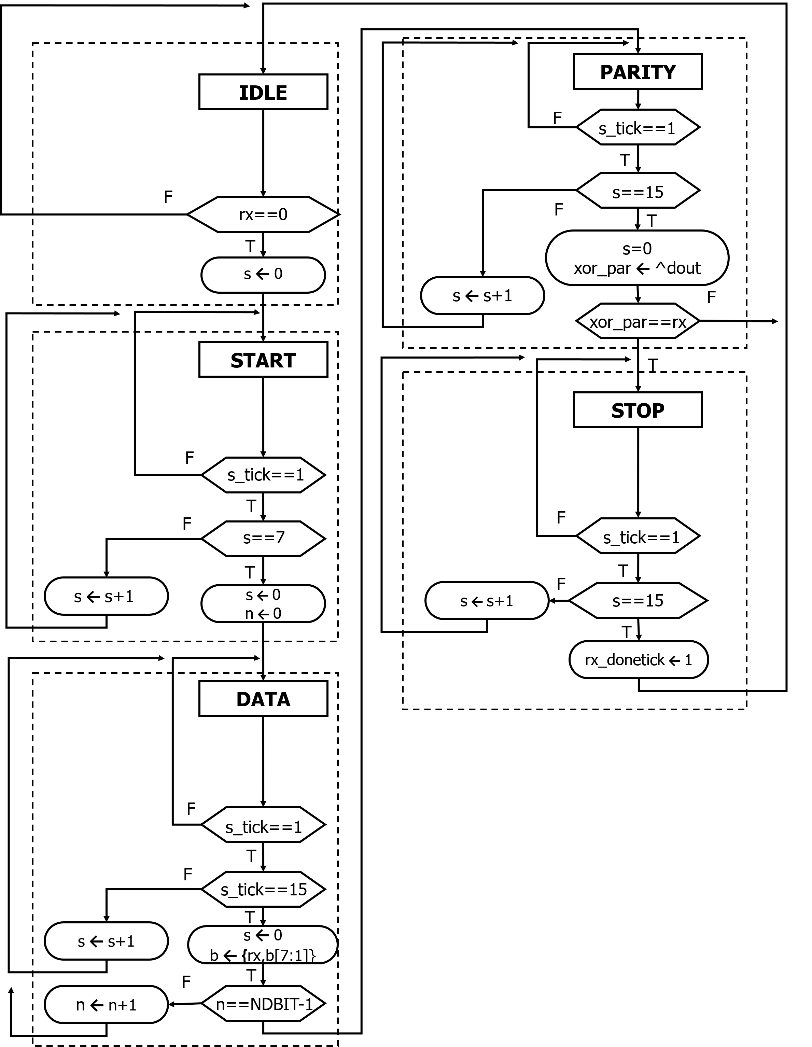
#### Khối thu RX



### Hình 3.7 Khối thu RX

Khối thu có chức năng nhận từng bit vào chân rx rồi đưa 8 bit ở chân dout.

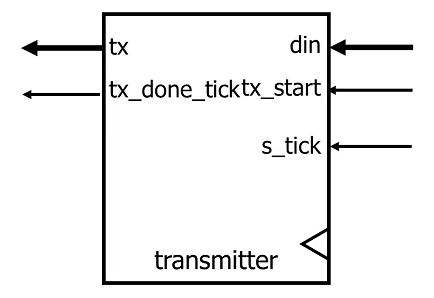
Sơ đồ thuật toán của khối thu được thực hiện như sau:



### Hình 3.8 Sơ đồ thuật toán khối RX

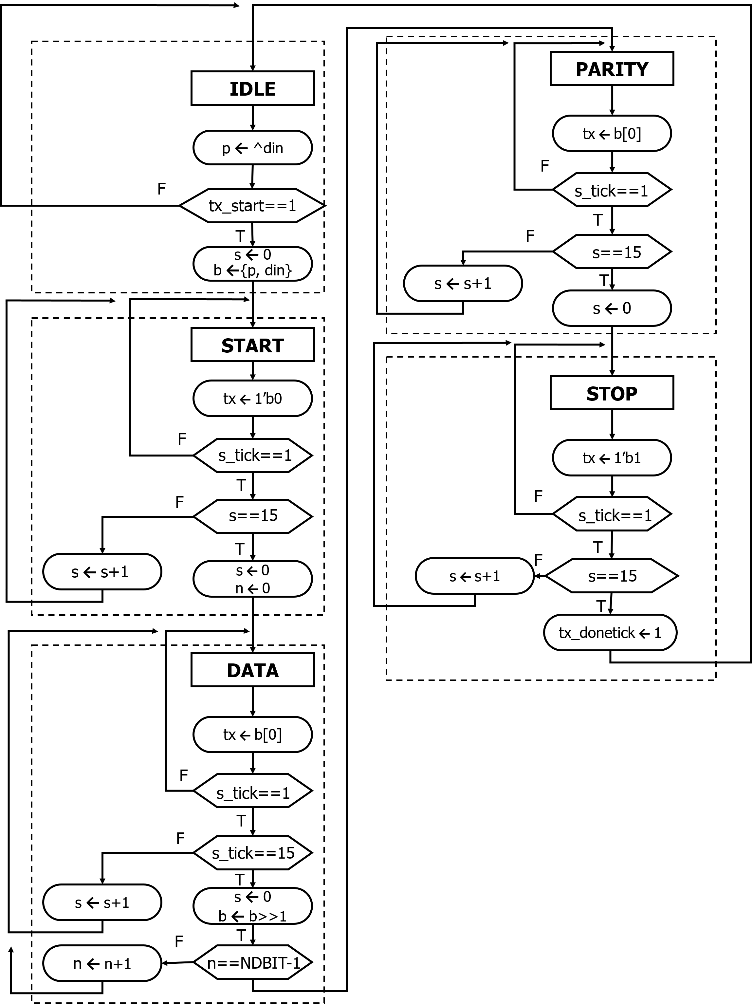
Trong đó có hai bộ đếm, được biểu diễn bởi các thanh ghi s và n. Thanh ghi s theo dõi số lượng tick lấy mẫu và đếm đến 7 ở bit bắt đầu, đến 15 ở bit dữ liệu và đến SB-TICK ở bit dừng. Thanh ghi n theo dõi số lượng bit dữ liệu nhận được ở trạng thái dữ liệu. Các bit truy xuất được chuyển vào và tập hợp lại trong thanh ghi b. Một tín hiệu rx-done-tick khẳng định quá trình nhận được hoàn thành.

#### Khối phát TX



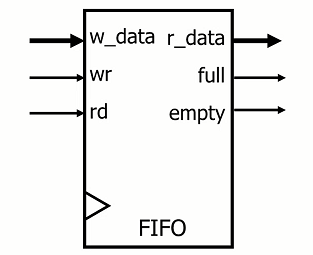
### Hình 3.9 Khối thu TX

Thuật toán của khối phát TX cũng gần tương tự như của RX. Sau khi nhận tín hiệu tx\_start, mỗi bit được chuyền ra sau mỗi 16 tick, tx\_done\_tick báo hiệu quá trình được hoàn thành.



### Hình 3.10 Sơ đồ thuật toán khối TX

#### Khối FIFO

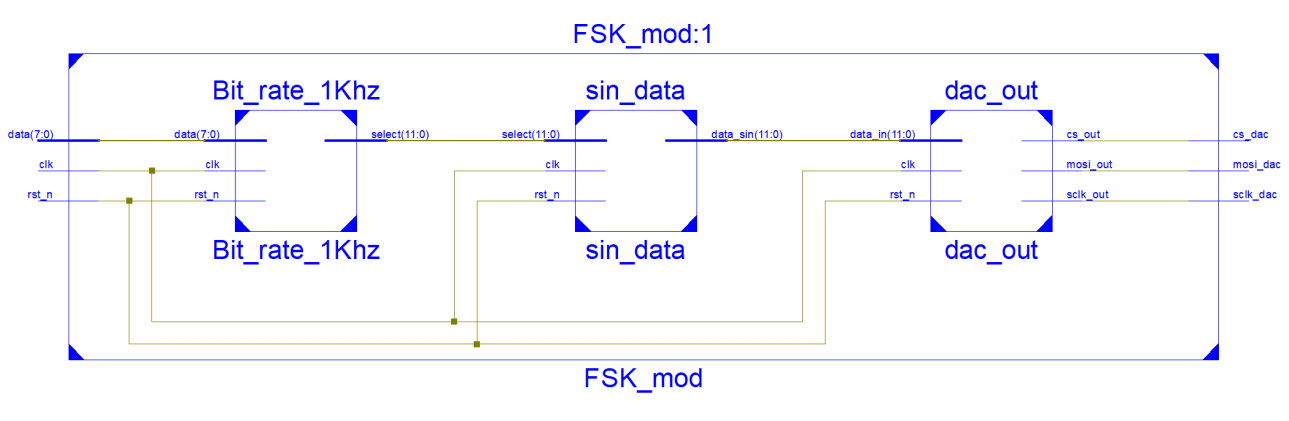


### Hình 3.11 Khối FIFO

Có chức năng đọc và ghi dữ liệu vào bộ đệm, giúp cung cấp không gian đệm giữa máy thu và hệ thống chính.

Tính hiệu 8 bit vào w\_data và ra ở r\_data

### FSK\_mod

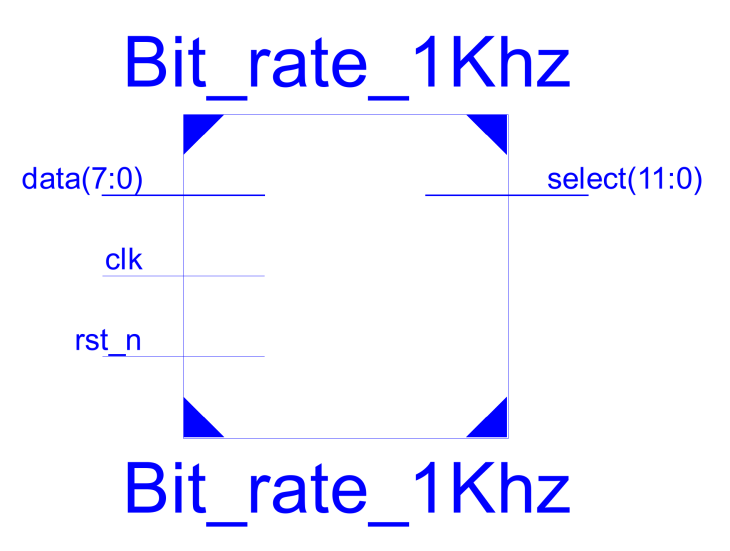


### Hình 3.12 Khối FSK\_Mod

Module FSK\_mod bao gồm 3 khối chính được kết nối với nhau như hình vẽ và có chức năng như sau:

* Khối Bit\_rate\_1Khz dùng để chuyển tín hiệu từ 8 bit song song sang 8 bit nối tiếp với tốc độ là 1Kbps, nối sang khối Sin\_data để điều khiển quá trình ghép sóng mang
* Khối Sin\_data nhận tín hiệu bit điều khiển từ khối Bit\_rate\_1Khz để ghép nối sóng mang theo bit đúng với lý thuyết FSK, bit 1 tần số bằng 16Khz, bit 0 tần số bằng 13 Khz
* Khối dac\_out giúp chuyển tín hiệu fsk\_signal sang DAC thông qua giao tiếp SPI

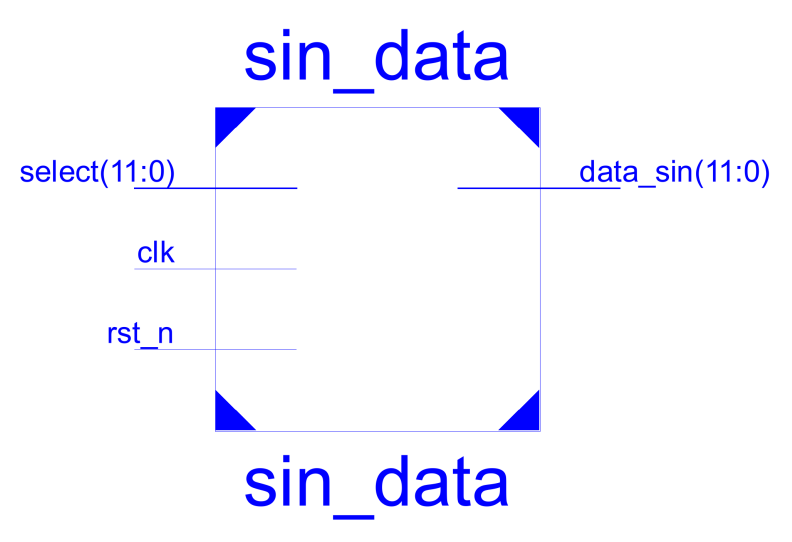
#### Khối Bit\_rate\_1Khz



Hình 3. 13 Sơ đồ chân khối Bit\_rate\_1Khz

#### Khối sin\_data

Tín hiệu sóng sin được điều chế dựa trên dòng bit đầu vào với bit 1 thì tần số sóng sin là 16 Khz và với bit 0 thì tần số sóng sin là 13 Khz



Hình 3.14 Sơ đồ chân khối sin\_data

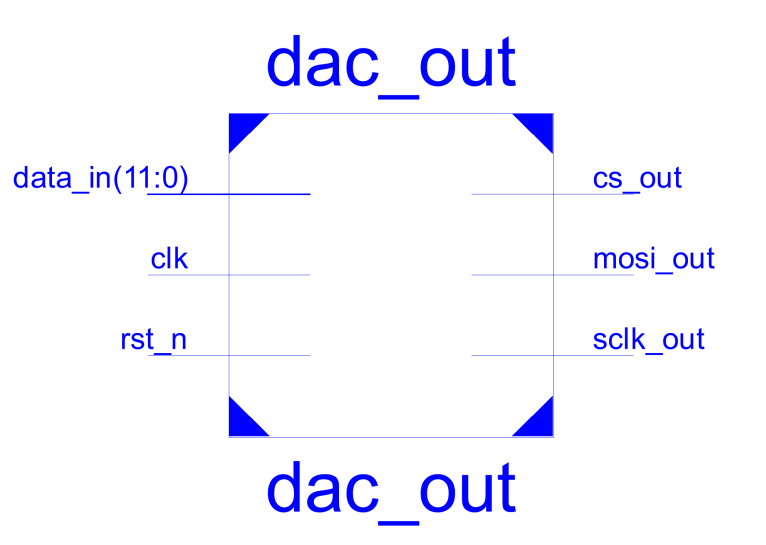
#### Khối dac\_out

Khối dac\_out giúp FPGA giao tiếp với DAC MCP 4921 để chuyển từ tín hiệu số sang tín hiệu tương tự thông qua giao tiếp SPI.

A close up of a keyboard

Description automatically generated

Hình 3. 15 SPI timing MCP 4921



Hình 3. 16 Sơ đồ chân khối dac\_out

# LINH KIỆN VÀ VI ĐIỀU KHIỂN

## Lựa chọn linh kiện

* Khả năng xử lý song song:

Chỉ cẩn 1 IC xung nhịp thấp có thể xử lý bài toán điều chế thay vì các dòng DSP, ARM cần có tốc độ làm việc cao hơn nhiều lần.

VD: FPGA có tốc độ clock 300Mhz đủ năng lực thiết kế phần số của máy thu phát với trung tần lên đến 60Mhz và băng thông 8Mhz thay vì dùng DSP phải cần đến bộ xử lý tốc độ 1Ghz, nếu dùng ARM thì tốc độ còn cao hơn nữa.

* Khả năng can thiệp sâu vào phần cứng:

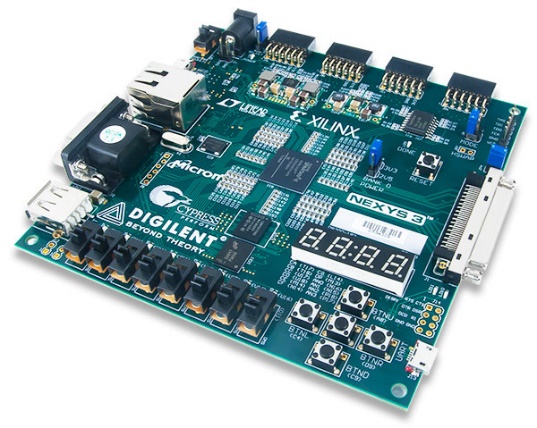
Thiết kế mạch số ở phần thu cần xử lý loại bỏ nhiễu, quyết định ngưỡng, mạch trễ, ... Nếu không can thiệp sâu vào phần cứng thì sự tối ưu của bộ giải điều chế sẽ giảm đi gây ảnh hưởng độ nhạy.

## Thông số và chức năng linh kiện

### Khối FSK\_Mod

Bảng 4.1 Thông số của các kit trên thị trường

|  |  |  |  |
| --- | --- | --- | --- |
|  | Kit FPGA Xilinx Spartan6+ module | Nexys A7 | Kit Nexys 3 |
| Giá thành tham khảo | $269.00 | $229.00 | $213.00 |
| Tần số đồng hồ | 50MHz | 450MHz | 500MHz + |
| Chip FPGA chính | Spartan6 XC6SLX9-TQG144 | Xilinx Arxtix-7 LX16 FPGA | Xilinx Spartan-6 LX16 FPGA​ |
| User I/O | 8 LED dán, 8 LED 7 đoạn 4 DIP switch | 16 công tắc, 16 đèn LED. Hai đèn LED RGB. Hai màn hình 4 chữ số 7 đoạn | 8 đèn LED, 5 nút, 8 công tắc trượt và màn hình 7 đoạn 4 chữ số |
| Kết nối | USB-UART | Cổng USB-UART và USB-HID  (cho chuột/ bàn phím) | Cổng USB-UART và USB-HID (cho chuột/ bàn phím) |



**Hình 4.1 Kit Nexys Xilinx Spartan 6 Nexys 3**

Các thông số của kit Nexy 3:

* Kit hoạt động ở tần số 100 MHZ
* 16 MB Cellular RAM
* 16MB SPI
* 10/100 Ethernet PHY
* USB2 cho chạy chương trình và truyền dữ liệu xfer
* USB-UART và USB-HID
* 8 bit VGA
* GPIO: 8 LEDs, 5 nút bấm, 8 công tắc và 4 LED 7 thanh
* 2278 slices, mỗi slices bao gồm 4 LUT 6 đầu vào và 8 Flip-flop
* 576kb cho khối RAM phục vụ xử lý nhanh
* 32 DSP

Đánh giá

* Lựa chọn Kit Nexys Xilinx Spartan 6 Nexys 3 vì phục vụ tốt cho nhu cầu của sinh viên, và đặc biệt là giá thành thấp
* Tần số hoạt động cao giúp cho việc xử lý trong các khối hoạt động nhanh

### Khối khuếch đại

#### Transistor BC547



Hình 4.2 Transistor BC547

BC547 là transistor BJT NPN đa năng được sử dụng trong các dự án điện tử giáo dục, thương mại. Nó được đóng gói TO- 92 và các dòng đầu ra tối đa mà transistor này có thể xử lý là 100mA. Transistor có độ lợi dòng điện một chiều rất tốt và độ nhiễu thấp do đó nó rất lý tưởng để sử dụng trong các giai đoạn khuếch đại tín hiệu. Điện áp bão hòa chỉ là 90mV cũng là một tín hiệu tốt để sử dụng nó như một công tắc

Chức năng: Có thể thay thế nhiều tran khác nên nó có thể sử dụng nhiều trong loại mạch điện tử như là chuyển đổi tải nhỏ trên điện áp và dòng đầu vào rất thấp cũng như khuếch đại âm thanh nhỏ và tín hiệu khác. Tần số chuyển tiếp tối đa của transistor là 300MHz vì vậy nó cũng sẽ hoạt động tốt trong các mạch RF dưới tần số 300MHz

#### Op Amp LF356N



Hình 4.3 Op Amp LF356N

Ưu điểm và chức năng :

* Hai ngõ vào đảo và không đảo cho phép khuếch đại được nguồn tín hiệu có tính đối xứng
* Ngõ ra chỉ khuếch đại sự sai lệch giữa hai tín hiệu ngõ vào nên Op-Ams có độ miễn nhiễu rất cao
* Khuếch đại cả những tín hiệu với biên độ chỉ vài chục mico Volt
* Tổng trở ngõ vào của Op-Amps rất lớn, cho phép mạch khuếch đại những nguồn tín hiệu có công suất bé

### Khối Tranducer

Bộ chuyển đổi này là thiết bị chuyển đổi năng lượng từ dạng này sang dạng khác. Cụ thể ở đây là chuyển từ tín hiệu điện sang tín hiệu âm thanh

Một số thông số kĩ thuật để đánh giá bộ chuyển đổi như: dải động học, độ ổn định, nhiều độ trễ

Đặc điểm nổi bật:

* Truyền tải âm thanh không có tạp âm
* Điện điện áp DC 5V và 2A
* Điện tiêu thụ là 0.5W
* Hỗ trợ âm thanh stereo 24bit S/ PDIF đầu vào
* Tỷ lệ mẫu 32/44.1/48/96Khz

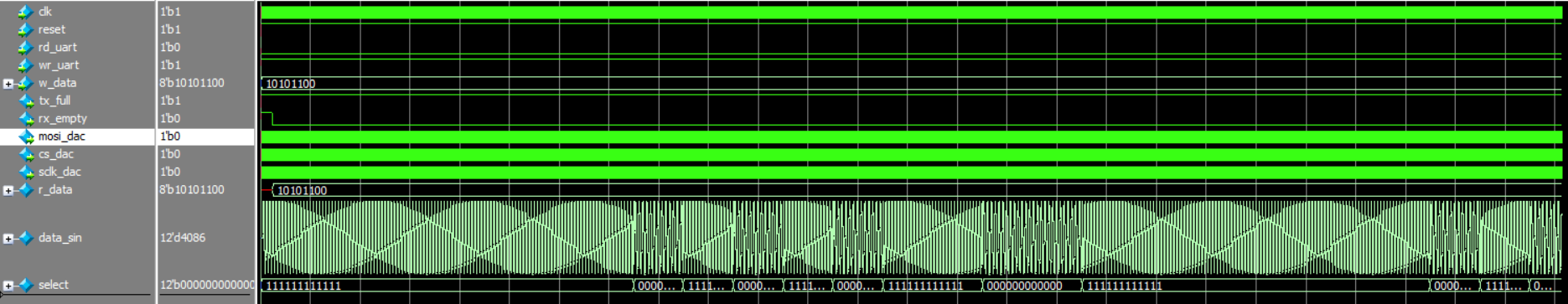
### Giá thành linh kiện.

**Bảng 4.2 Bảng giá thành linh kiện**

|  |  |
| --- | --- |
| Linh kiện | Giá thành |
| Kit Nexys 3 | 213$ = 4.899.000 đồng |
| Transducer | 130,000 đồng |
| Transistor BC547 | 1000 đồng |
| Op-Amp LF356N | 22000 đồng |
| Tụ điện, điện trở | 2000 đồng |
| Tổng chi phí | * 1. ồng |

# Kết quả

Do tình hình dịch bệnh phức tạp, bản thân một số thành viên của nhóm lại ở trong vùng dịch và là F1 nên nhóm chúng em chưa thể đến trường, phòng Lab để tiếp cận Kit và linh kiện. Do đó nhóm chúng em xin được đưa kết quả mô phỏng tín hiệu thay thế.



Hình 5. 1 Wareform

Từ hình ảnh mô phỏng có thể thấy tín hiệu vào w\_data và tín hiệu r\_data của UART là giống nhau, qua đó khẳng định khối chuẩn giao tiếp UART là đúng.

Tín hiệu data\_sin khi đi qua khối FSK\_Mod cũng ra kết quả với tần số cao ứng với bit 1, tần số thấp ứng với bit 0, đúng như lý thuyết.

# KẾT LUẬN

Trong quá trình làm việc nhóm, từ những hôm thảo luận trên lớp để tìm ra ý tưởng cho sản phẩm rồi qua quá trình tìm hiểu nhu cầu của người sử dụng và thị trường để lên ý tưởng, thực hiện phương án, nhóm đã lựa chọn việc thiết kế hệ thống điều chế truyền thông FSK Theo chuẩn uart sử dụng transducer trên FPGA

Qua những buổi họp nhóm , bọn em đã học được cách thiết kế ra một sản phẩm theo đúng quy trình, từ việc lựa chọn đề tài, lên ý tưởng, kiểm tra tính khả thi của ý tưởng cho đến các bước tạo demo và thử nghiệm sản phẩm, từ đó hoàn thiện sản phẩm của mình. Mong rằng thiết kế của bọn em sẽ trở nên hữu dụng trong tương lai.

Trong thời gian thực hiện bài tập lớn cộng với kiến thức còn nhiều hạn chế và dịch bệnh Covid-19 diễn biến phức tạp nên trong báo cáo không tránh khỏi những thiếu xót, nhóm chúng em mong nhận được sự đánh giá, góp ý của thầy và các bạn trong suốt quá trình thảo luận cũng như thiết kế sản phẩm.

Nhóm em xin gửi lời cảm ơn đến thầy Hàn Huy Dũng và toàn bộ các bạn sinh viên trong lớp đã có những nhận xét đối với bản thiết kế, qua đó để sản phẩm có thể được hoàn thiện tốt hơn trong tương lai.

# Tài liệu tham khảo

[[1]](https://online.bpostel.com/index.php/bpostel/article/view/180102/426) FREQUENCY SHIFT KEYING DEMODULATORS FOR LOW-POWER FPGA APPLICATIONS, RILEY T. HARRINGTON B.S, Kansas State University, 2011, [https://core.ac.uk/download/pdf/77979986.pdf](https://online.bpostel.com/index.php/bpostel/article/view/180102/426)

[2] FPGA-based FSK/PSK modulation, Ahmed Asim Ghouri, Embedded Strings Inc.EDN (December 11, 2014)

[*https://www.design-reuse.com/articles/36109/fpga-based-fsk-psk-modulation.html*](https://www.design-reuse.com/articles/36109/fpga-based-fsk-psk-modulation.html)

[1] *Pong P. Chu. (2008) FPGA Prototyping by Verilog Examples: Xilinx Spartan-3 Version*

[2] <https://en.wikipedia.org/wiki/Frequency-shift_keying>

[3] Kennedy, G.; Davis, B. (1992). Electronic Communication Systems (4th ed.). McGraw-Hill International. [ISBN](https://en.wikipedia.org/wiki/International_Standard_Book_Number) [978-0-07-112672-4](https://en.wikipedia.org/wiki/Special:BookSources/978-0-07-112672-4)., p 509

[4]<https://reference.digilentinc.com/reference/programmable-logic/nexys-3/reference-manual>

[4] <https://www.microchip.com/wwwproducts/en/MCP4921>

[5] Kỹ sư N. Đ. Thiện, (2019). *NGHIÊN CỨU VÀ TRIỂN KHAI KHỐI ĐIỀU CHẾ FSK VÀ BỘ LỌC SỐ TRÊN FPGA.* Đại học Bách khoa Hà Nội.

Embedded Strings Inc.EDN (December 11, 2014)