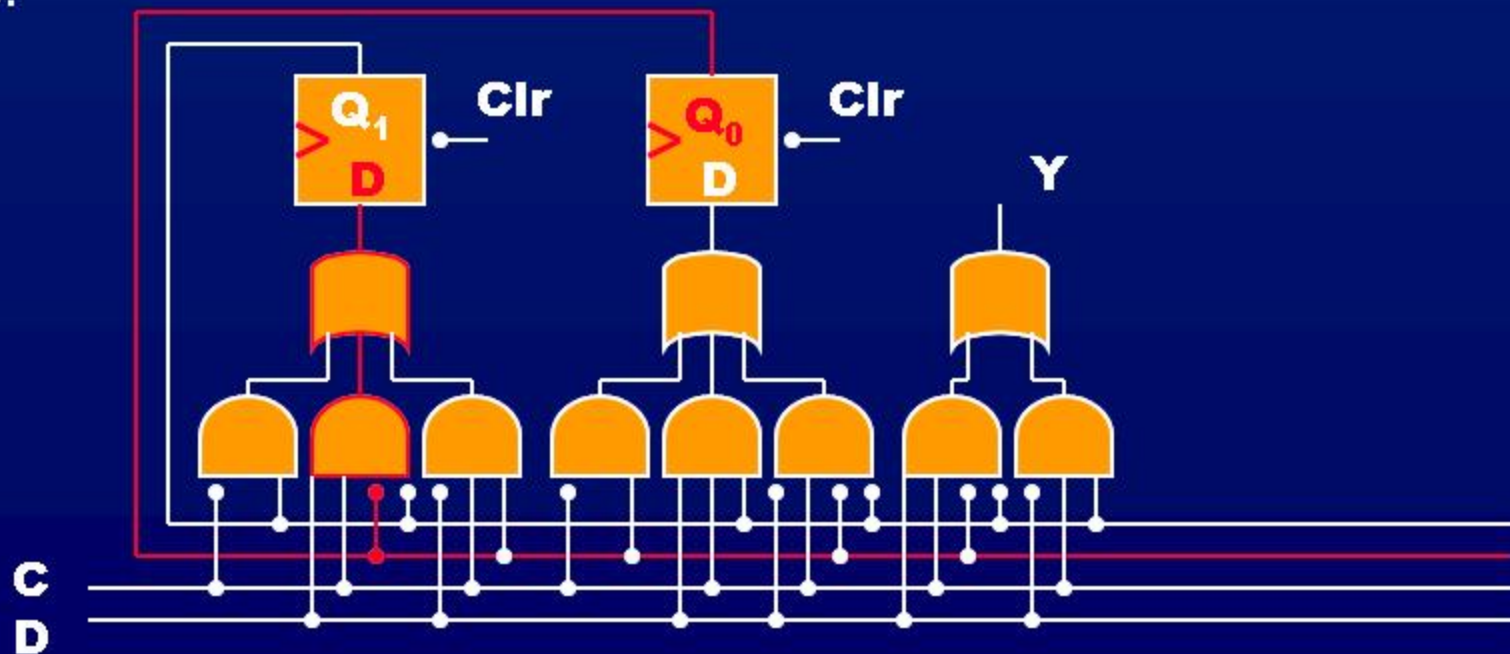


# Bước 6: Phân tích tín hiệu theo thời gian

- Xác định tần số tối đa:
  - ❑ Tần số tối đa =  $1/\text{trễ của đường dài nhất}$
  - ❑ Đường dài nhất là đường có trễ mạch tổ hợp lớn nhất giữa hai sườn xung nhịp
  - ❑ Ví dụ:



# Thiết kế mạch số tuần tự

## Các mạch số tuần tự cơ bản

---

- Thanh ghi (Registers)
- Thanh ghi dịch (Shift registers)
- Bộ đếm (Counter)
- Tập thanh ghi (Register files)
- Cấu trúc hàng đợi vào sau ra trước LIFO (stack)
- Cấu trúc hàng đợi vào trước ra trước FIFO

## 5. Ngôn ngữ VHDL

# Giới thiệu về VHDL

- VHDL = VHSIC Hardware Description Language
- VHSIC = Very High Speed Integrated Circuit
- Là ngôn ngữ lập trình dùng để mô tả hoạt động của hệ thống số
- Được quy định trong chuẩn IEEE 1076 từ năm 1983
- Các ngôn ngữ mô tả phần cứng khác:
  - Verilog
  - Abel



**library** IEEE; -- Su dung thu vien chuan IEEE

**use** IEEE.STD\_LOGIC\_1164.ALL; --Su dung tat ca cac thanh phan trong goi STD\_LOGIC\_1164

**entity** hex2led **is**

**Port** ( **HEX** : **in** std\_logic\_vector(3 **downto** 0);

**LED** : **out** std\_logic\_vector(6 **downto** 0));

**end** hex2led;

-- Khai bao hoat dong cua hex2Led

**architecture** Behavioral **of** hex2led **is**

**begin**

**with** HEX **SElect**

    LED<= "1111001" **when** "0001", --1

        "0100100" **when** "0010", --2

        "0110000" **when** "0011", --3

        "0011001" **when** "0100", --4

        "0010010" **when** "0101", --5

        "0000010" **when** "0110", --6

        "1111000" **when** "0111", --7

        "0000000" **when** "1000", --8

        "0010000" **when** "1001", --9

        "0001000" **when** "1010", --A

        "0000011" **when** "1011", --b

        "1000110" **when** "1100", --C

        "0100001" **when** "1101", --d

        "0000110" **when** "1110", --E

        "0001110" **when** "1111", --F

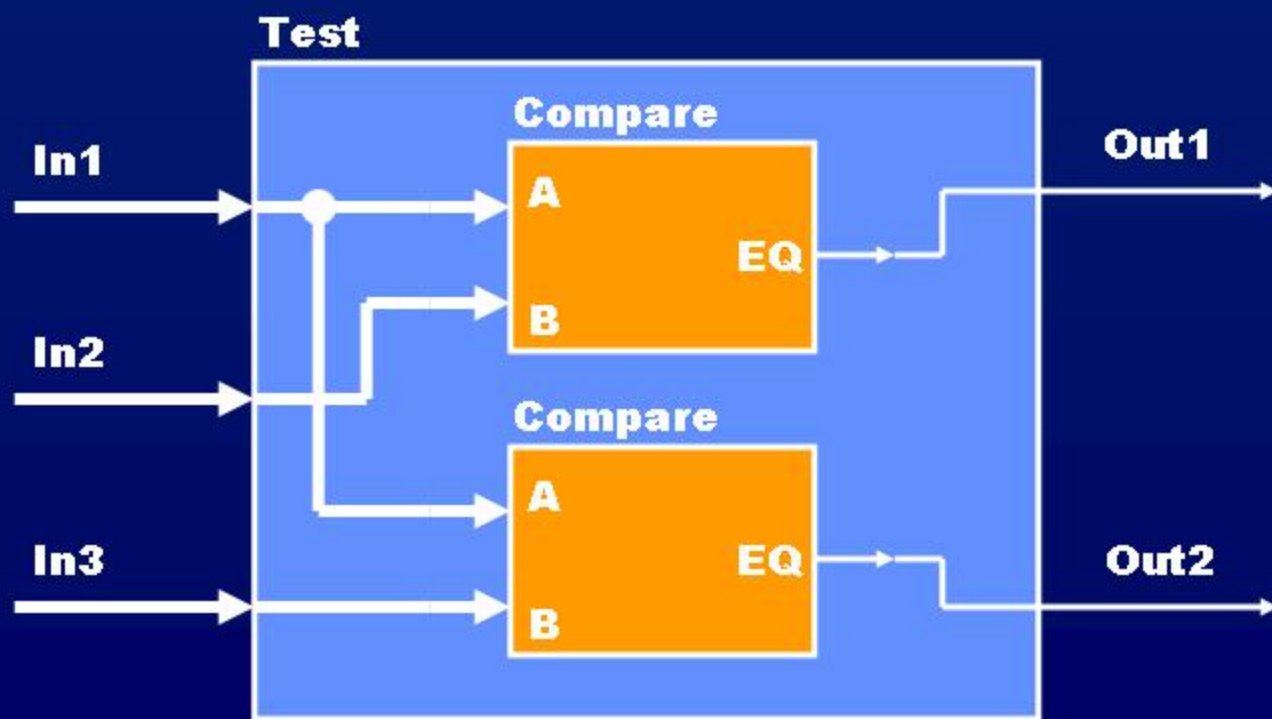
        "1000000" **when** **others**; --0

**end** Behavioral;

# Giới thiệu về VHDL

## Ví dụ 1

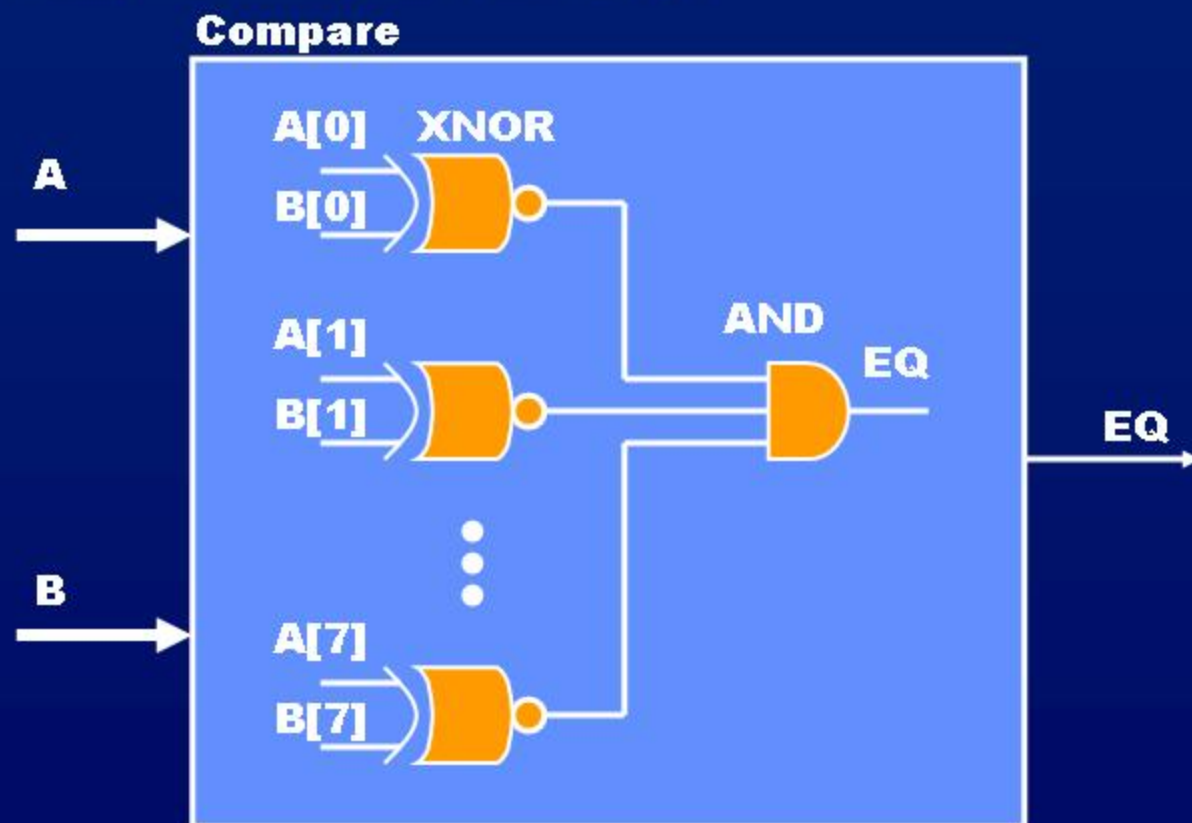
- Thiết kế mạch 'Test' với 3 đầu vào 8-bit (In1, In2, In3) và hai đầu ra 1 bit (Out1, Out2). Out1='1' khi In1=In2 và Out2='1' khi In1 = In3



# Giới thiệu về VHDL

## Ví dụ 1

Thiết kế khối compare dùng mạch tổ hợp



# Giới thiệu về VHDL

## Ví dụ 1

- Thiết kế khối compare dùng VHDL

‘Entity’ xác định giao diện với bên ngoài của khối cần thiết kế

- ~~Eight~~ bit comparator

**entity** Compare **is**

**port** (A,B: **in** bit\_vector(0 to 7);

EQ: **out** bit);

**end** Compare;

đầu vào và ra được gọi là port

**architecture** Behav1 **of** Compare **is**

**begin**

EQ <= ‘1’ **when** (A=B) **else** ‘0’;

**end** Behav1;

‘Architecture’ miêu tả hoạt động và cấu trúc bên trong của khối cần thiết kế

Chú ý:

- Một entity có thể có nhiều architecture, mỗi architecture là một cách thể hiện khác nhau của cùng một chức năng
- Các Ports là vector có chiều: vào (in), ra (out), hoặc cả vào cả ra (inout)



# Giới thiệu về VHDL

## Component và Instantiation

- Biểu diễn Test bằng VHDL

**entity Test is**

**port**( In1,In2,In3: **in** bit\_vector(0 to 7);  
Out1,Out2: **out** bit);

**end Test;**

**architecture Struct1 of Test is**

**component Comparator is**

**port**( X,Y: **in** bit\_vector(0 to 7);  
Z: **out** bit);

**end component;**

**begin**

Compare1: Comparator **port map** (X=>In1, Y=>In2, Z=>Out1);

Compare2: Comparator **port map** (X=>In1,Y=>In3,Z=>Out2);

**end Struct1;**

2 bản copy của cùng một  
component  
'Comparator'

Chú ý:

- Hai bản comparator chạy song song với nhau !!!
- Đây là architecture miêu tả cấu trúc của entity Test

# Giới thiệu về VHDL

## Cấu hình (Configuration)

- Khi một entity có nhiều architectures, ta sẽ xử dụng architecture nào?
- Làm thế nào để gắn ‘Components’ với ‘Entities’?
  - **Configuration information: architecture selection**
  - **and component-entity binding**

**configuration Build1 of Test is**

**for Struct1**

**for Compare1: Comparator use entity Compare(Behav1)**

**port map (A => X, B => Y, EQ => Z);**

**end for;**

**for others: Comparator use entity Compare(Behav1)**

**port map (A => X, B => Y, EQ => Z);**

**end for;**

**end for;**

**end Build1;**