

Hệ thống số lập trình

3 tín chỉ (11 buổi Lý thuyết + 11 buổi Bài tập + Thực hành)

TS. Trần Mạnh Cường

Mục đích môn học

1. Tổng quan về công nghệ IC khả trình và các ứng dụng.
2. Quy trình thiết kế sử dụng IC khả trình
3. Thiết kế hệ thống số đơn giản đến phức tạp sử dụng ngôn ngữ mô tả phần cứng Verilog HDL/VHDL
4. Thực hành ứng dụng trên KIT Spartan-3E của hãng Xilinx

Nội dung môn học

1. Nhắc lại kiến thức điện tử số (1 LT + 0 BT)
2. Giới thiệu công nghệ IC lập trình được (1 LT + 0 BT)
3. Ngôn ngữ lập trình Verilog HDL (2 LT + 2 BT)
4. Thiết kế mạch số thông dụng (2 LT + 2 BT)
5. Công cụ là quy trình thiết kế với IC. lập trình được (2 LT + 2 BT)
6. Thực hành thiết kế trên FPGA (3 LT + 3 BT)
7. Kiểm tra 1 buổi + dự phòng 1 buổi

Tài liệu tham khảo

- Slide Bài giảng
- Tống Văn On. Nguyên Lý Mạch Tích Hợp - Tập 1,2: ASIC Lập Trình Được (Ấn Bản Dành Cho Sinh Viên). Nhà xuất bản Thống Kê, 2005.
- Tống Văn On. Thiết Kế Mạch Số Với VHDL & Verilog - Tập 1,2. Nhà xuất bản Lao động-Xã hội, 2007.
- FPGA prototyping by VHDL examples SpartanTM-3 Version, Pong P. Chu, WILEY- INTERSCIENCE, 2008.
- Verilog HDL-A guide to Digital Design and Synthesis (Samir palnitkar)
- Các nguồn tài liệu trên Internet về :
 - ASIC, FPGA
 - Verilog HDL
 - Spartan 3E, phần mềm thiết kế ISE của Xilinx

<http://www.xilinx.com/support.html>

Đánh giá môn học

- Điểm quá trình: trọng số 0,3 (Kiểm tra + Bài tập + Bảo vệ thiết kế thực hành)
- Điểm thi cuối kì: trọng số 0,7
- Hình thức thi: Tự luận có sử dụng tài liệu

Nhắc lại kiến thức điện tử số

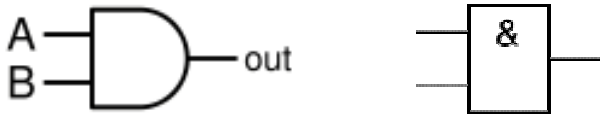
- Biến và Hàm logic
- Các phần tử logic cơ bản
- Hệ logic tổ hợp
- Hệ logic dãy

Biến và Hàm logic

- Biến logic: đặc trưng cho trạng thái logic của các đối tượng. Được biểu diễn bởi các ký tự, trạng thái logic có giá trị là 0 hoặc 1
- Hàm logic: là biểu diễn của nhóm các biến logic, liên hệ với nhau thông qua các phép toán logic
- 3 phép toán cơ bản
 - và (AND)
 - hoặc (OR)
 - phủ định - đảo (NOT)

Phép AND và cổng AND (AND gate)

- Chức năng: thực hiện phép toán AND logic
- Cổng AND 2 đầu vào

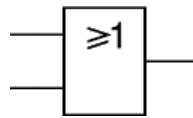
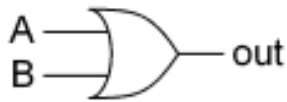


- Bảng chân lý
- Biểu thức $out = A.B$

A	B	out
0	0	0
0	1	0
1	0	0
1	1	1

Phép OR và cổng OR (OR gate)

- Chức năng: thực hiện phép toán OR logic
- Cổng OR 2 đầu vào

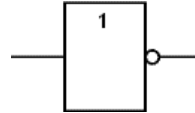
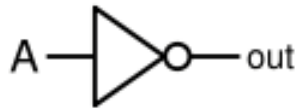


- Bảng chân lý
- Biểu thức $out = A + B$

A	B	out
0	0	0
0	1	1
1	0	1
1	1	1

Phép NOT và cổng NOT (NOT gate)

- Chức năng: thực hiện phép toán NOT logic
- Cổng NOT chỉ có 1 đầu vào

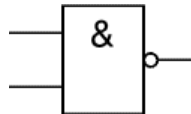
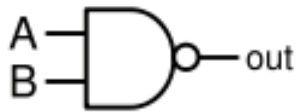


- Bảng chân lý
- Biểu thức $out = \overline{A}$

A	out
0	1
1	0

Phép VÀ ĐẢO (NAND gate)

- Chức năng: thực hiện phép toán NOT logic của phép toán AND
- Cổng NAND có 2 đầu vào



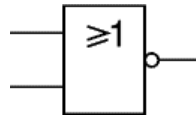
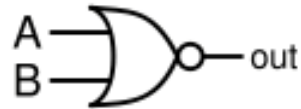
- Bảng chân lý
- Biểu thức

$$out = \overline{A.B}$$

A	B	out
0	0	1
0	1	1
1	0	1
1	1	0

Phép HOẶC ĐẢO (NOR gate)

- Chức năng: thực hiện phép toán NOT logic của phép toán OR
- Cổng NAND có 2 đầu vào



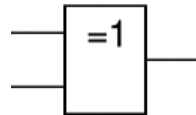
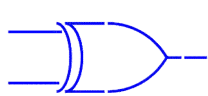
- Bảng chân lý
- Biểu thức

$$out = \overline{A + B}$$

A	B	out
0	0	1
0	1	0
1	0	0
1	1	0

Phép XOR (XOR gate)

- Chức năng: thực hiện phép toán HOẶC CÓ LOẠI TRỪ, đầu ra bằng 0 khi có đầu vào giống nhau
- Cổng XOR có 2 đầu vào

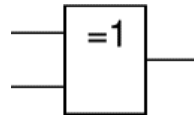


- Bảng chân lý
- Biểu thức $out = \bar{A}.B + A.\bar{B} = A \oplus B$

A	B	out
0	0	0
0	1	1
1	0	1
1	1	0

Phép XNOR (XNOR gate)

- Chức năng: thực hiện phép toán NOT của phép XOR, đầu ra bằng 1 khi có đầu vào giống nhau
- Cổng XOR có 2 đầu vào



- Bảng chân lý

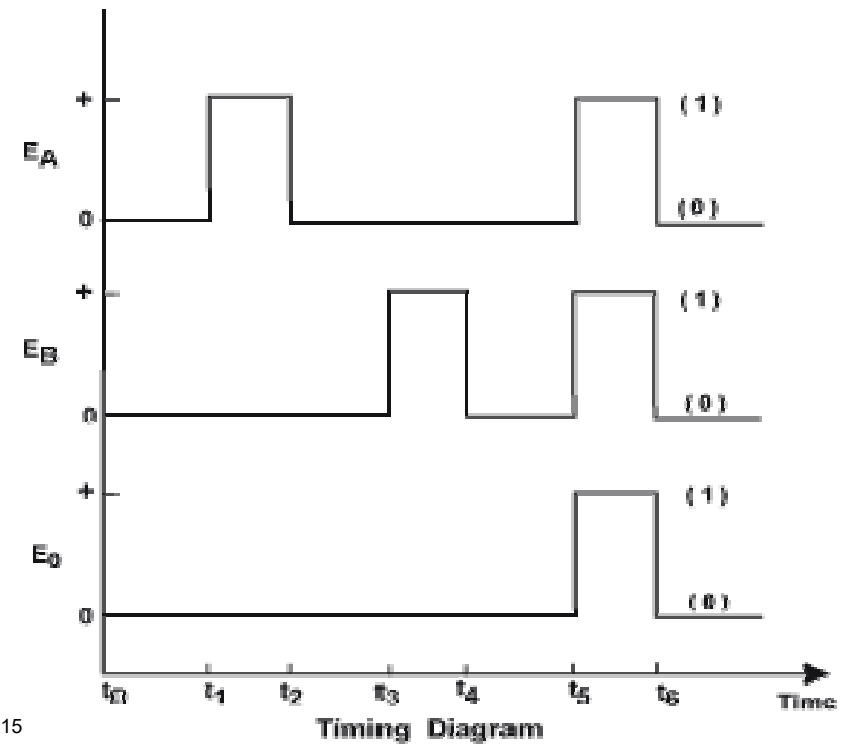
- Biểu thức $out = \overline{A \oplus B} = A.B + \overline{A}.\overline{B}$

A	B	out
0	0	1
0	1	0
1	0	0
1	1	1

Câu hỏi

- Cho biểu đồ thời gian sau, hãy cho biết từ biểu đồ thời gian biểu diễn hoạt động của cổng nào ?

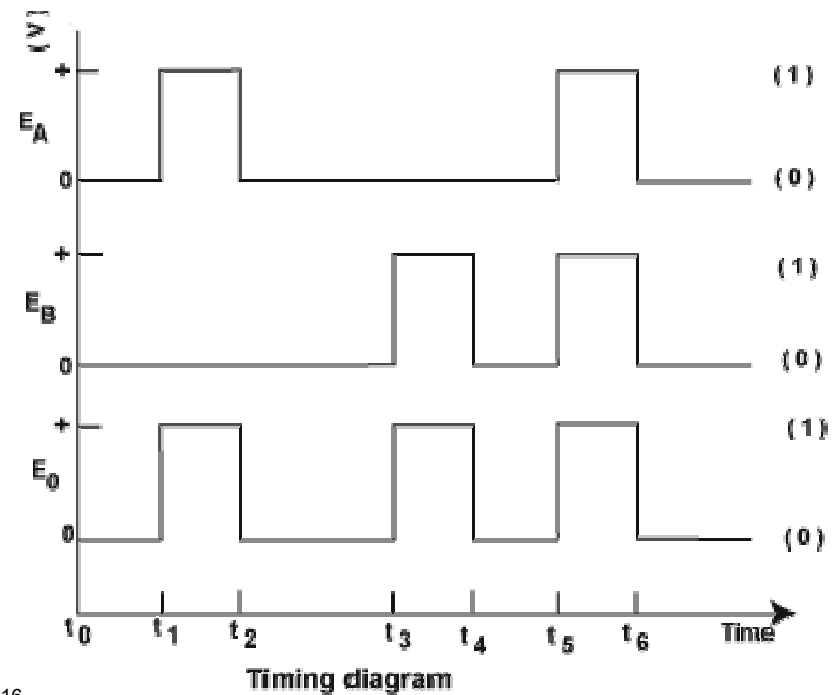
$$E_0(E_A, E_B) = ?$$



Câu hỏi

- Cho biểu đồ thời gian sau, hãy cho biết từ biểu đồ thời gian biểu diễn hoạt động của cổng nào ?

$$E_0(E_A, E_B) = ?$$

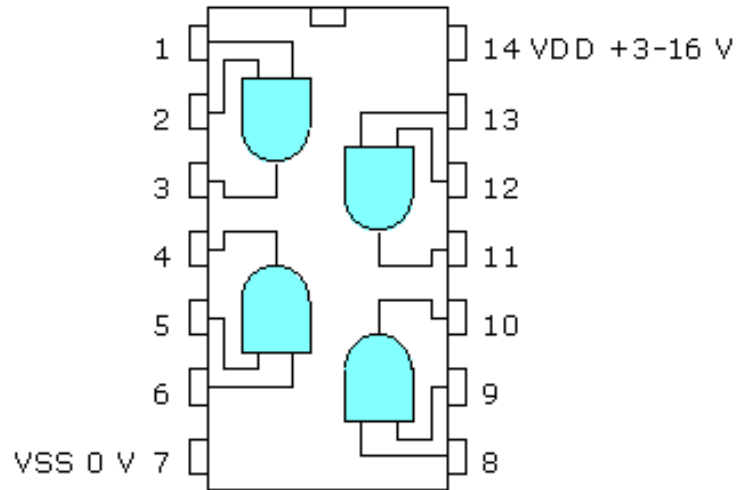


Các phần tử logic cơ bản

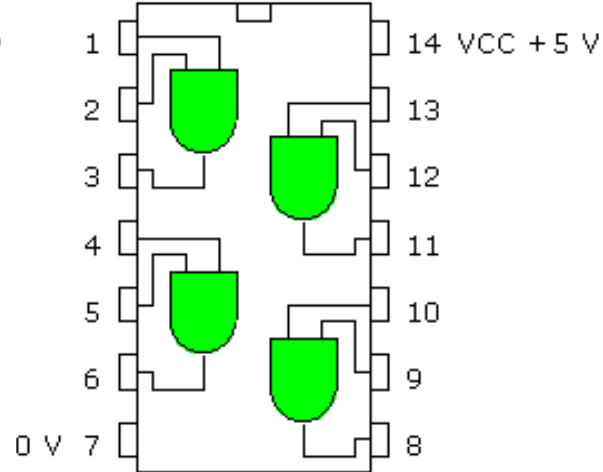
- AND: 74LS08, 4081, 4073,...
- OR: 74LS32, 4071, 4072,...
- NOT: 74LS04/05
- NAND: 74LS00, 4011, 4012,...
- NOR: 74LS02, 4000, 4001, 4002,...
- XOR: 74LS136, 4070/4030,...
- XNOR: 74LS266, 4077,...

AND IC

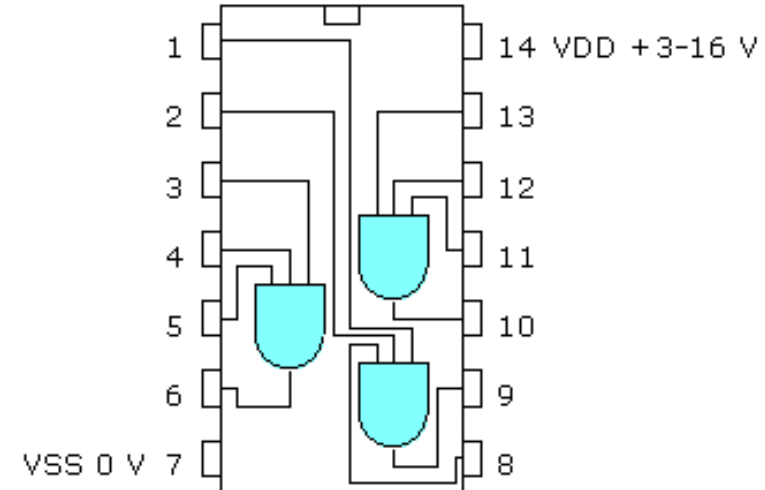
4081



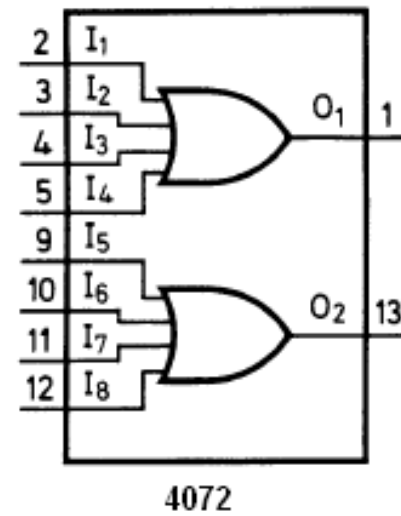
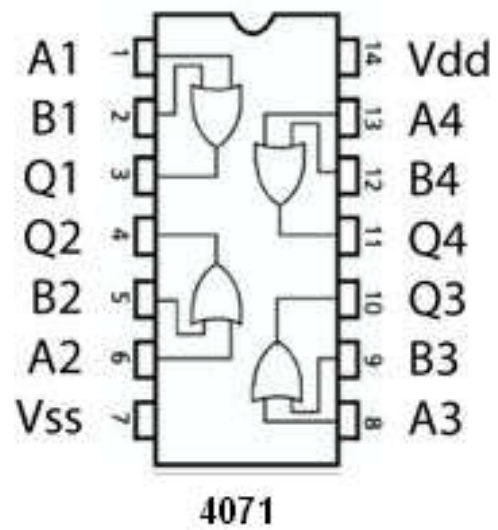
74LS08



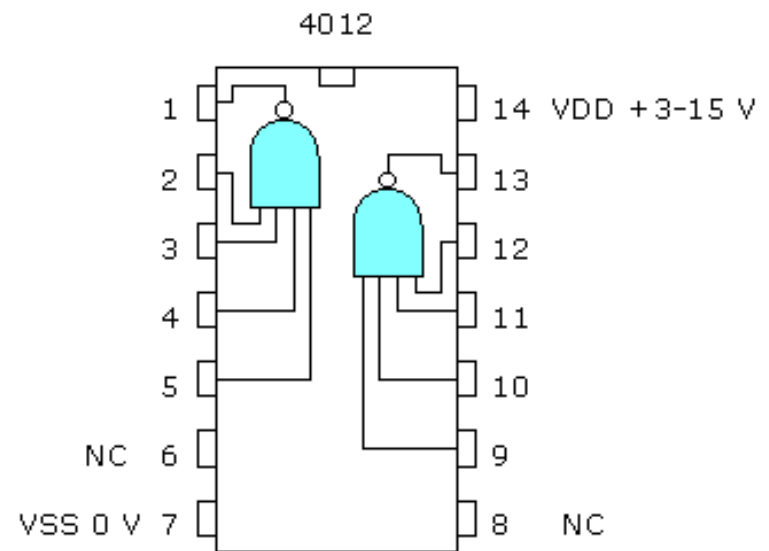
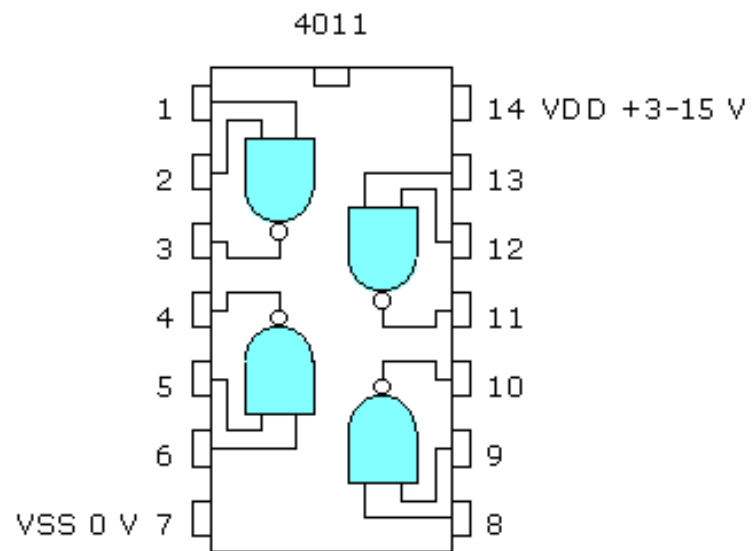
4073



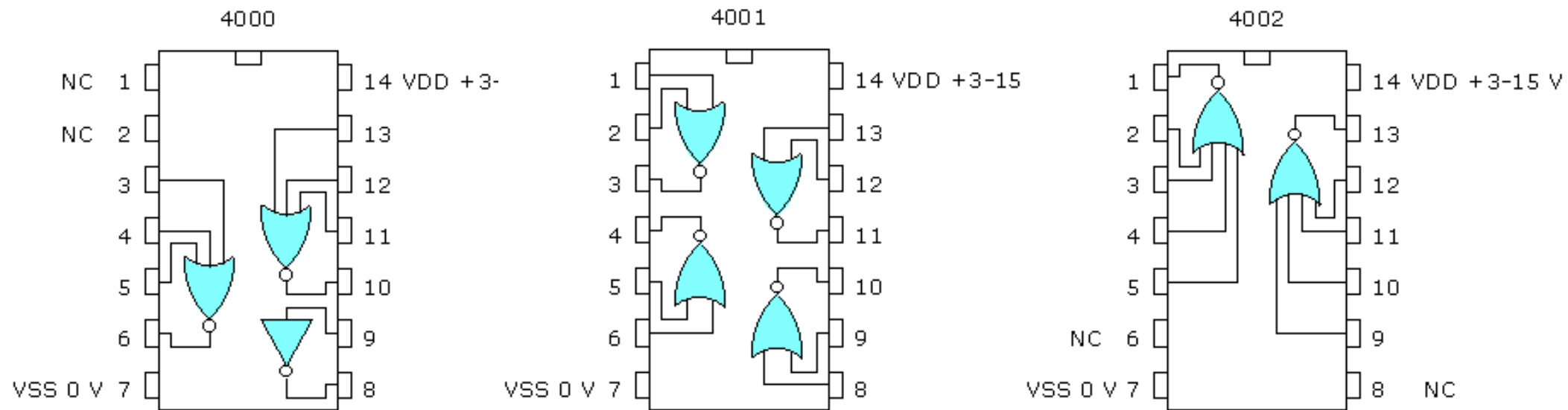
OR IC



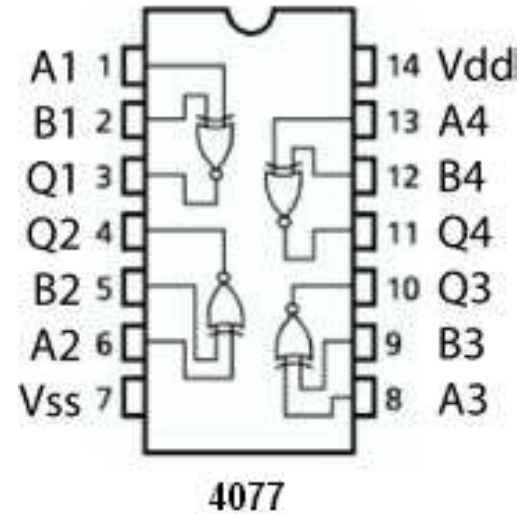
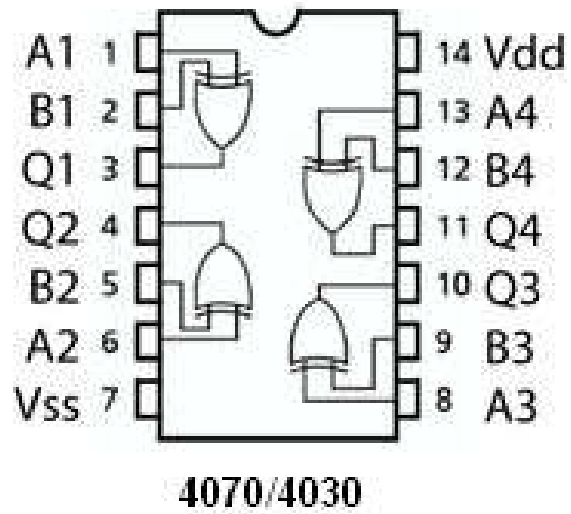
NAND IC



NOR IC



XOR và XNOR IC



Câu hỏi

- Biểu diễn các phần tử logic 2 đầu vào AND, OR và phần tử logic 1 đầu vào NOT sử dụng
 - Chỉ dùng các phần tử NAND
 - Chỉ dùng các phần tử NOR

Hệ tổ hợp

- là hệ mà tín hiệu ra chỉ phụ thuộc vào tín hiệu vào tại thời điểm hiện tại
- còn được gọi là hệ không có nhớ
- có thể được tổng hợp từ các phần tử logic cơ bản

Một số hệ tổ hợp cơ bản

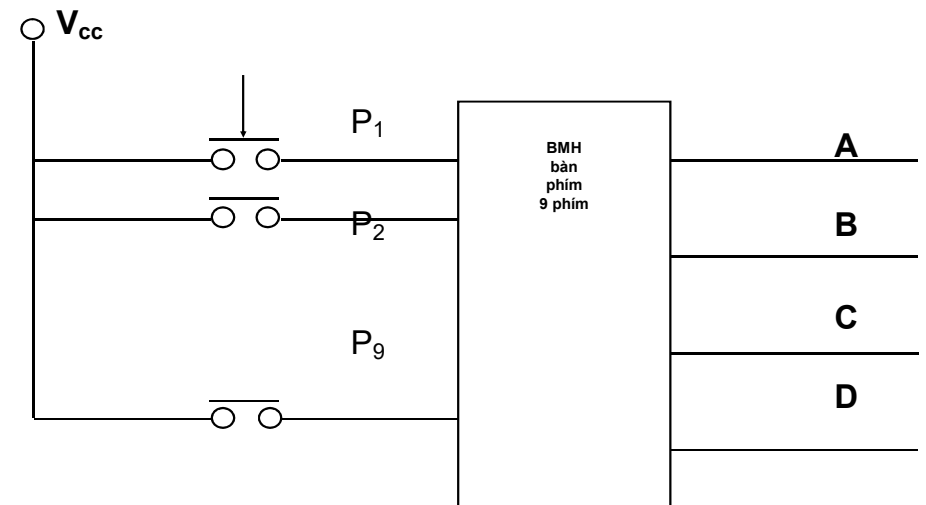
- bộ mã hoá
- bộ giải mã
- bộ chọn kênh
- bộ phân kênh
- các mạch số học

Ví dụ - bộ mã hoá bàn phím

- Mã hoá bàn phím
 - Mỗi phím được gán với 1 mã riêng biệt
 - Khi phím được tác động, bộ mã hoá sẽ đưa ra ở đầu ra mã tương ứng
- Bàn phím gồm 9 nút ấn, thiết kế bộ mã hoá với giả sử tại một thời điểm chỉ có một phím được ấn

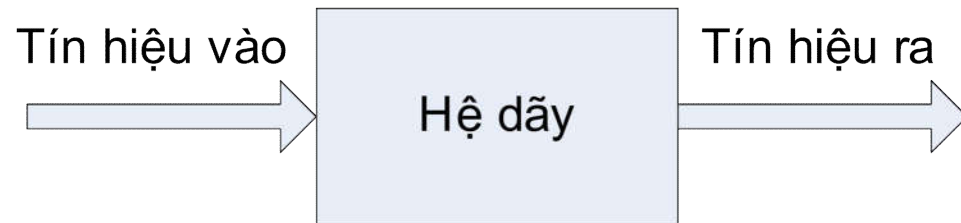
Ví dụ - bộ mã hoá bàn phím

- Sơ đồ
 - Có 9 phím nên cần phải sử dụng mã 4 bit để mã hoá
 - Bộ giải mã sẽ có 9 đầu vào và 4 bit đầu ra



Hệ logic dãy

- Là mạch có tính chất nhớ
- Trạng thái đầu ra của mạch phụ thuộc trạng thái của các biến đầu vào + trạng thái hiện tại của mạch



- 2 loại mô hình
 - Mealy
 - Moore

Trạng thái

Mô hình Mealy

- Mô hình mô tả hệ dãy thông qua 5 tham số
 - $X = \{x_1, x_2, x_3, \dots, x_n\}$ - tập n tín hiệu đầu vào
 - $Y = \{y_1, y_2, y_3, \dots, y_l\}$ - tập l tín hiệu đầu ra
 - $S = \{S_1, S_2, S_3, \dots, S_m\}$ - tập m trạng thái trong hệ
 - $F_S(S, X)$ - hàm biến đổi trạng thái
 - $F_Y(S, X)$ - hàm tính trạng thái đầu ra - trong mô hình Mealy, F_Y phụ thuộc S và X

Mô hình Moore

- Mô hình mô tả hệ dãy thông qua 5 tham số
 - $X = \{x_1, x_2, x_3, \dots, x_n\}$ - tập n tín hiệu đầu vào
 - $Y = \{y_1, y_2, y_3, \dots, y_l\}$ - tập l tín hiệu đầu ra
 - $S = \{S_1, S_2, S_3, \dots, S_m\}$ - tập m trạng thái trong hệ
 - $F_S(S, X)$ - hàm biến đổi trạng thái
 - $F_Y(S)$ - hàm tính trạng thái đầu ra - trong mô hình Moore, F_Y chỉ phụ thuộc S

Bảng chuyển trạng thái

- Mô hình Mealy

S	X		
	X_1	...	X_n
S_1	$F_S(S_1, X_1), F_Y(S_1, X_1)$...	$F_S(S_1, X_n), F_Y(S_1, X_n)$
...
S_m	$F_S(S_m, X_1), F_Y(S_m, X_1)$...	$F_S(S_m, X_n), F_Y(S_m, X_n)$

- Mô hình Moore

S	X			Y
	X_1	...	X_n	
S_1	$F_S(S_1, X_1)$...	$F_S(S_1, X_n)$	$F_Y(S_1)$
...
S_m	$F_S(S_m, X_1)$...	$F_S(S_m, X_n)$	$F_Y(S_m)$

Bảng chuyển trạng thái

- Mô hình Mealy

S	X		
	X_1	...	X_n
S_1	$F_S(S_1, X_1), F_Y(S_1, X_1)$...	$F_S(S_1, X_n), F_Y(S_1, X_n)$
...
S_m	$F_S(S_m, X_1), F_Y(S_m, X_1)$...	$F_S(S_m, X_n), F_Y(S_m, X_n)$

- Mô hình Moore

S	X			Y
	X_1	...	X_n	
S_1	$F_S(S_1, X_1)$...	$F_S(S_1, X_n)$	$F_Y(S_1)$
...
S_m	$F_S(S_m, X_1)$...	$F_S(S_m, X_n)$	$F_Y(S_m)$

Ví dụ

- Sử dụng mô hình Mealy và mô hình Moore để mô tả hệ dãy thực hiện phép tính cộng

	t_4	t_3	t_2	t_1	t_0
A	0	1	1	0	0
B	0	1	1	1	0
S	1	1	0	1	0

Ví dụ

- Mô hình Mealy
 - $X = \{00, 01, 10, 11\}$ - do có 2 đầu vào A và B
 - $Y = \{0,1\}$ - có 1 đầu ra S (sum)
 - $S = \{s_0, s_1\}$ - có 2 trạng thái s_0 là không nhớ, s_1 là có nhớ
 - $F_S(S,X)$
 - $F_Y(S,X)$

$$F_S(s_0,00) = s_0$$

$$F_S(s_0,01) = s_0$$

$$F_S(s_0,00) = 0$$

$$F_S(s_0,01) = 1$$

$$F_S(s_0,10) = s_0$$

$$F_S(s_0,11) = s_1$$

$$F_S(s_0,10) = 1$$

$$F_S(s_0,11) = 0$$

$$F_S(s_1,00) = s_0$$

$$F_S(s_1,01) = s_1$$

$$F_S(s_1,00) = 1$$

$$F_S(s_1,01) = 0$$

$$F_S(s_1,10) = s_1$$

$$F_S(s_1,11) = s_1$$

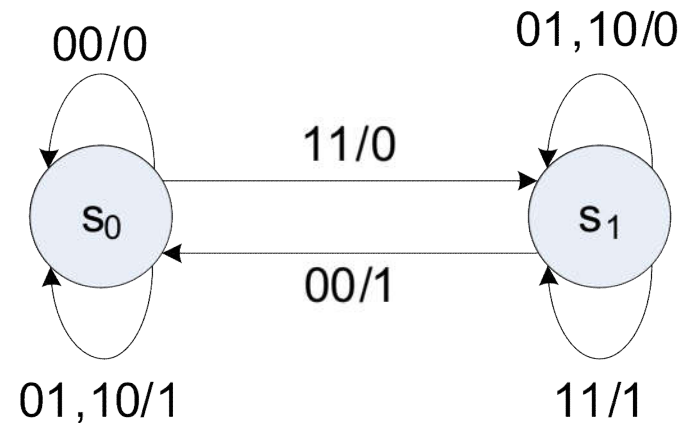
$$F_S(s_1,10) = 0$$

$$F_S(s_1,11) = 1$$

Ví dụ

- Mô hình Mealy
 - Bảng chuyển trạng thái
 - Đồ hình chuyển trạng thái

S	X			
	00	01	10	11
s_0	$s_0, 0$	$s_0, 1$	$s_0, 1$	$s_1, 0$
s_1	$s_0, 1$	$s_1, 0$	$s_1, 0$	$s_1, 1$



Ví dụ

- Mô hình Moore
 - $X = \{00, 01, 10, 11\}$ - do có 2 đầu vào A và B
 - $Y = \{0, 1\}$ - có 1 đầu ra S (sum)
 - $S = \{s_{00}, s_{01}, s_{10}, s_{11}\}$ - có 4 trạng thái

i - là có nhớ (1) hoặc không nhớ (0)

s_{ij} j - là tín hiệu ra

Ví dụ

- Mô hình Moore

- $F_S(S,X)$

$$F_S(s_{00},00) = s_{00}$$

$$F_S(s_{00},10) = s_{01}$$

$$F_S(s_{01},00) = s_{00}$$

$$F_S(s_{01},10) = s_{01}$$

$$F_S(s_{10},00) = s_{01}$$

$$F_S(s_{10},10) = s_{10}$$

$$F_S(s_{11},00) = s_{01}$$

$$F_S(s_{11},10) = s_{10}$$

$$F_S(s_{00},01) = s_{01}$$

$$F_S(s_{00},11) = s_{10}$$

$$F_S(s_{01},01) = s_{01}$$

$$F_S(s_{01},11) = s_{10}$$

$$F_S(s_{10},01) = s_{10}$$

$$F_S(s_{10},11) = s_{11}$$

$$F_S(s_{11},01) = s_{10}$$

$$F_S(s_{11},11) = s_{11}$$

- $F_Y(S)$

$$F_Y(s_{00}) = 0$$

$$F_Y(s_{01}) = 0$$

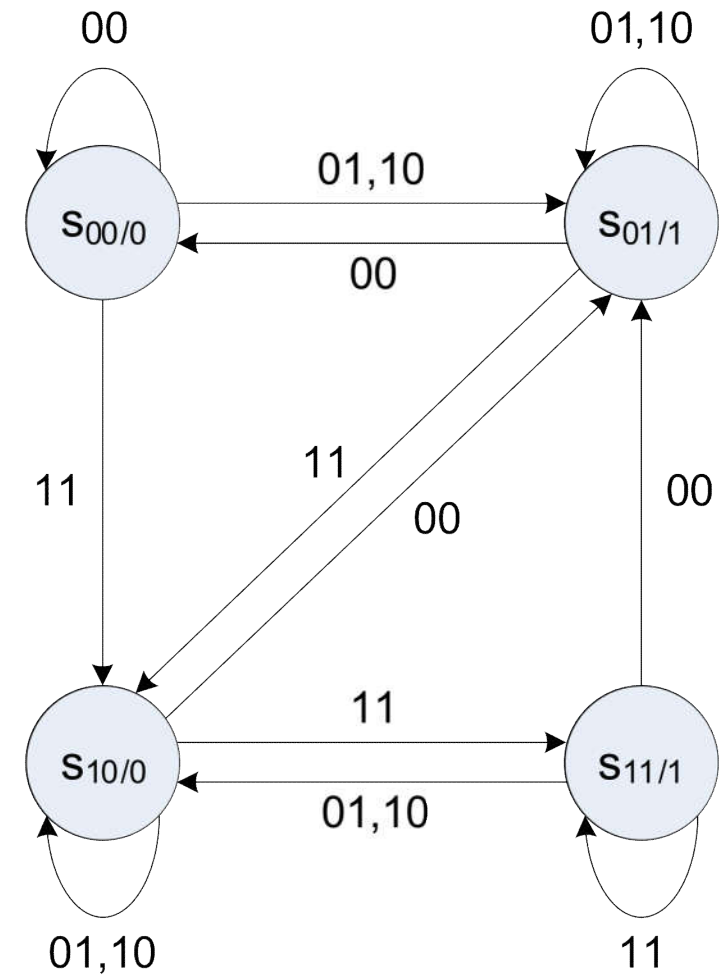
$$F_Y(s_{10}) = 1$$

$$F_Y(s_{11}) = 1$$

Ví dụ

- Mô hình Moore
 - Bảng chuyển trạng thái
 - Đồ hình chuyển trạng thái

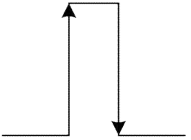
S	X				Y
	00	01	10	11	
s ₀₀	s ₀₀	s ₀₁	s ₀₁	s ₁₀	0
s ₀₁	s ₀₀	s ₀₁	s ₀₁	s ₁₀	1
s ₁₀	s ₀₁	s ₁₀	s ₁₀	s ₁₁	0
s ₁₁	s ₀₁	s ₁₀	s ₁₀	s ₁₁	1



Phần tử cơ bản của mạch logic dãy

- Phần tử nhớ - trigger
- Đầu ra của trigger là trạng thái của nó
- Phân loại theo trạng thái làm việc
 - Không đồng bộ: trạng thái đầu ra chỉ phụ thuộc trạng thái tín hiệu vào
 - Đồng bộ: trạng thái đầu ra phụ thuộc trạng thái tín hiệu đầu vào và tín hiệu đồng bộ

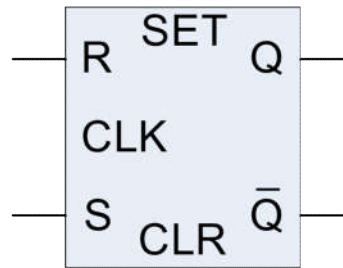
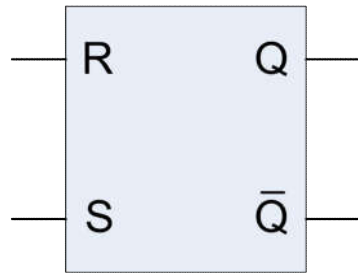
Các kiểu đồng bộ

Kiểu đồng bộ		Trạng thái làm việc	Trạng thái nghỉ - giữ nguyên trạng thái
Theo mức	Mức cao	Tín hiệu đồng bộ ở mức cao H	Tín hiệu đồng bộ ở mức thấp L
	Mức thấp	Tín hiệu đồng bộ ở mức thấp L	Tín hiệu đồng bộ ở mức cao H
Theo sườn	Sườn lên	Xung sườn lên của tín hiệu đồng bộ	trường hợp còn lại
	Sườn xuống	Xung sườn xuống của tín hiệu đồng bộ	trường hợp còn lại
Theo xung đồng bộ	 Đồng bộ kiểu xung	Có xung	Không có xung

Các loại trigger

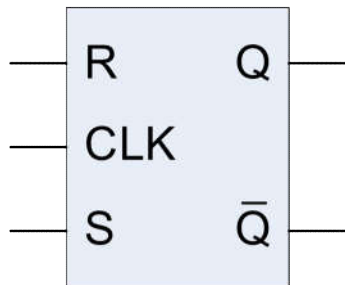
- RS - Set - Reset
- D - Delay
- JK - Jordan và Kelly
- T - Toggle

RS trigger

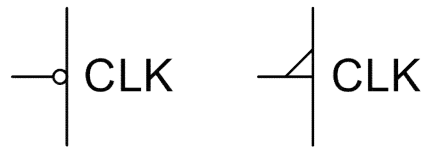


RS		00	01	11	10
q	0	0	1	-	0
	1	1	1	-	0
		nhớ	thiết lập	không xác định	xóa

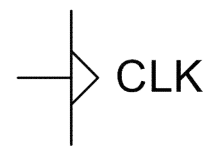
$$Q = S + q\bar{R}$$



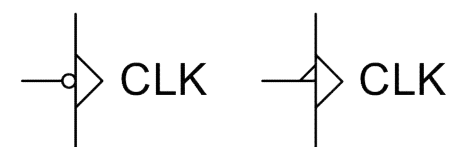
Đồng bộ mức cao



Đồng bộ mức thấp



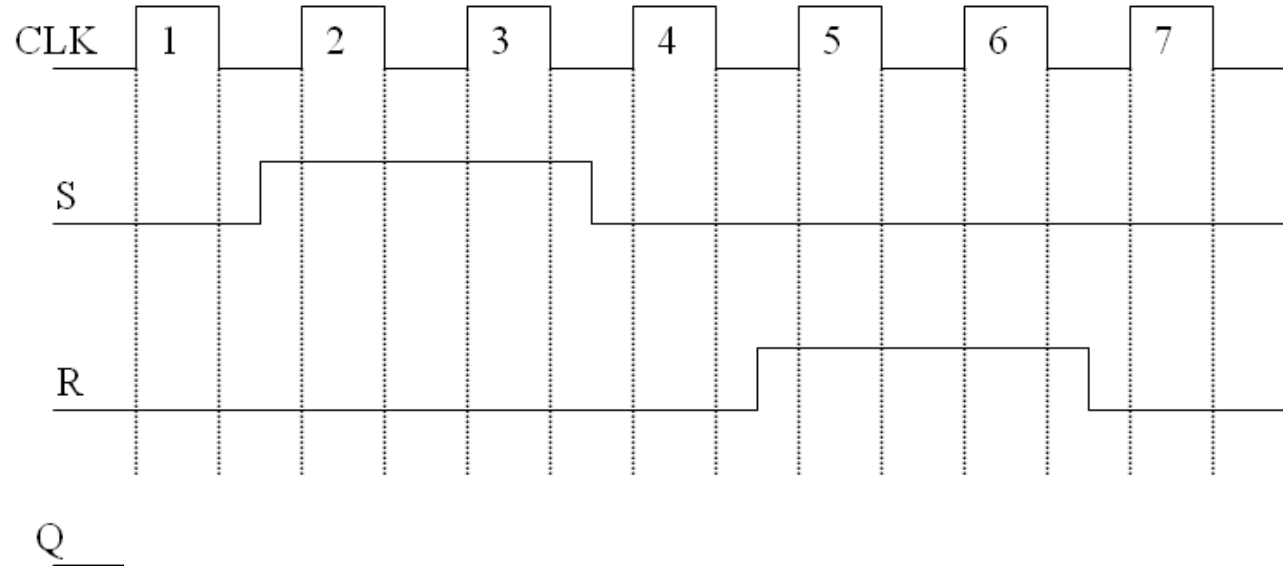
Đồng bộ sườn dương



Đồng bộ sườn âm

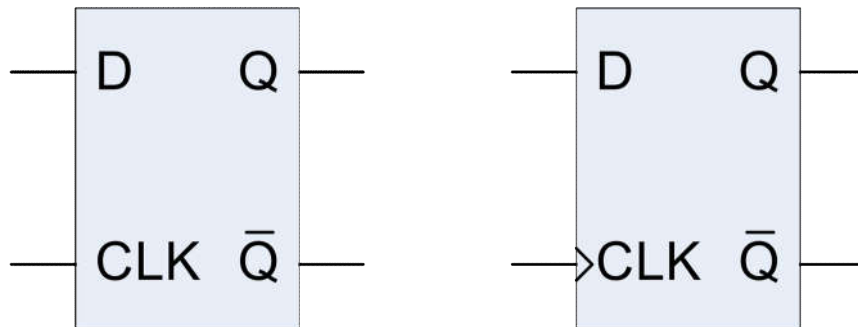
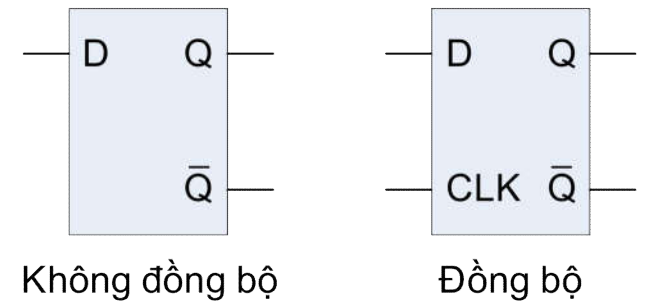
RS trigger

- Cho Trigger RS đồng bộ mức cao và đồ thị các tín hiệu R, S như hình vẽ. Hãy vẽ đồ thị tín



D trigger

- D trigger có 1 đầu vào D và hoạt động ở chế độ đồng bộ và không đồng bộ
- Chỉ xét D trigger ở chế độ đồng bộ
 - Theo mức - chốt D (Latch)
 - Theo sườn - xúc phát sườn (edge trigger)

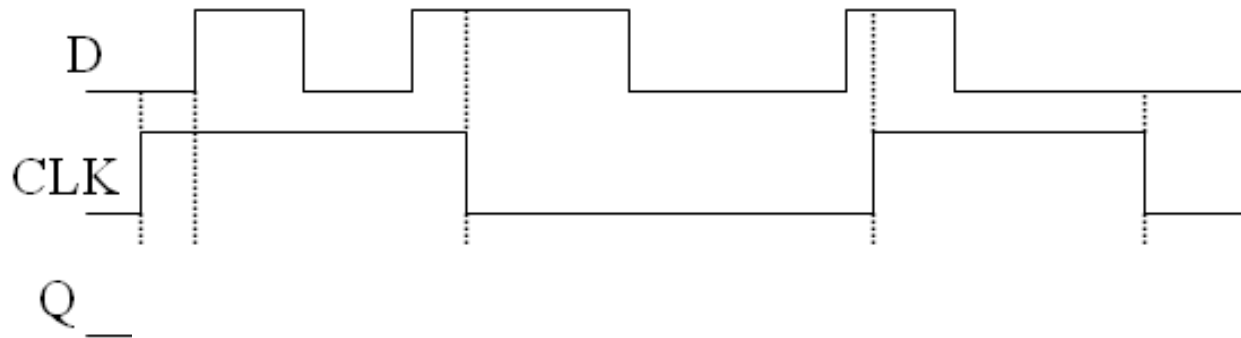


		D	
		0	1
q	0	0	1
	1	0	1

$$Q = D$$

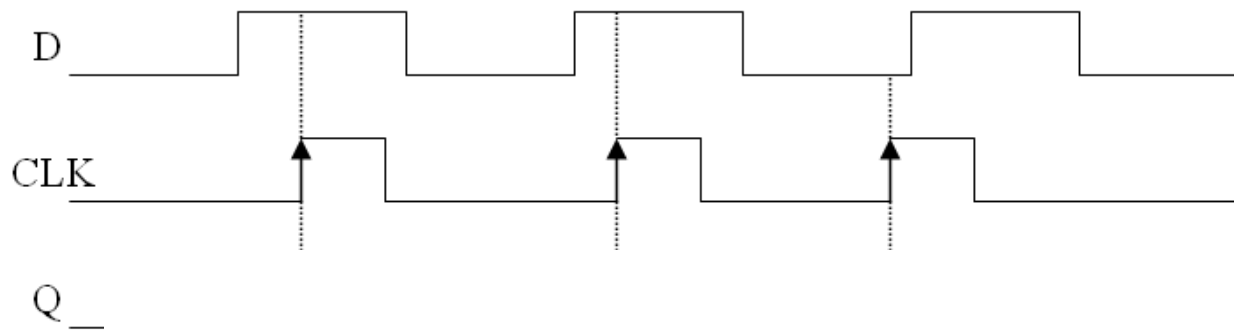
D trigger

- Cho chốt D kích hoạt mức cao. Hãy vẽ tín hiệu ra Q tương ứng trên cùng trục thời gian với tín hiệu vào D.



D trigger

- Cho chốt D kích hoạt sườn dương. Hãy vẽ tín hiệu ra Q tương ứng trên cùng trục thời gian với tín hiệu vào D.



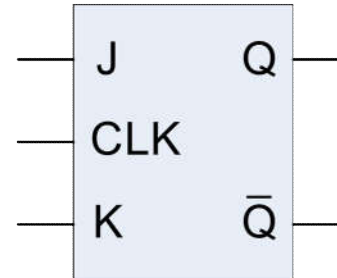
JK trigger

- JK trigger chỉ hoạt động ở chế độ đồng bộ

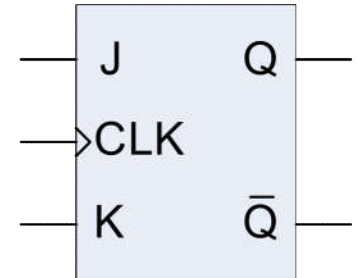
q \ JK				
	00	01	11	10
0	0	0	1	1
1	1	0	0	1
	nhớ	xóa	lật	thiết lập

J ~ S
K ~ R

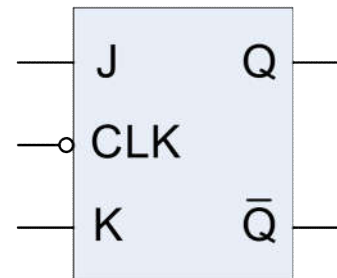
$$Q = \bar{q}J + q\bar{K}$$



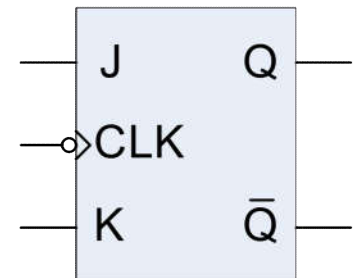
Tích cực mức cao



Tích cực sườn dương



Tích cực mức thấp



Tích cực sườn âm

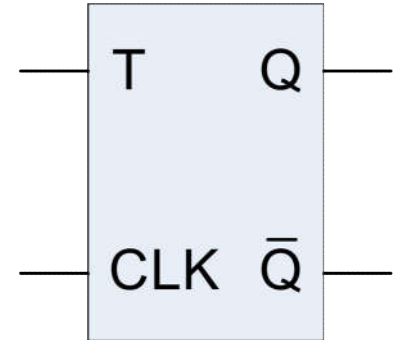
T trigger

- T trigger chỉ hoạt động ở chế độ đồng bộ

q \ T	0	1
0	0	1
1	1	0

nhớ lật

$$Q = \bar{q}T + q\bar{T} = q \oplus T$$



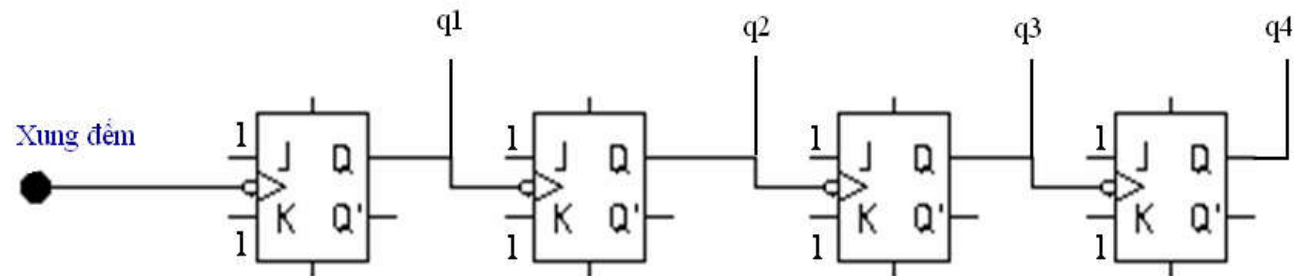
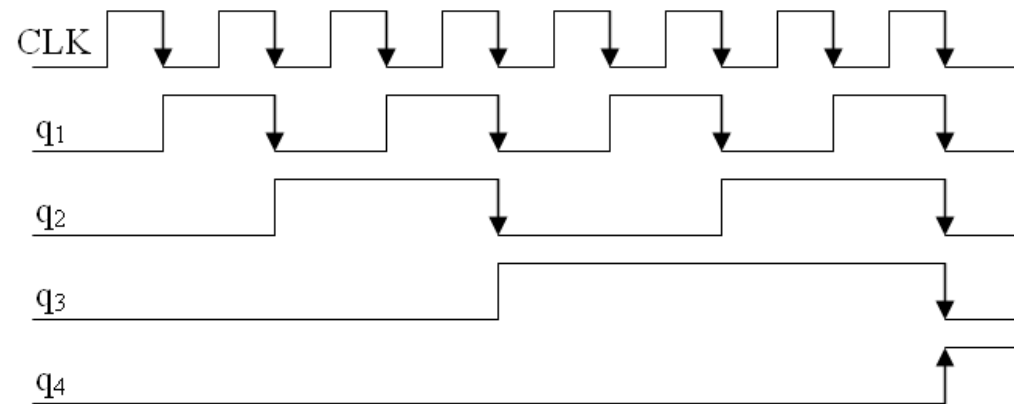
Một số ứng dụng hệ dãy

- Bộ đếm - đếm xung, bộ chia tần
 - Module n nếu đếm n xung từ 0 đến $n-1$
 - Đếm không đồng bộ và đếm đồng bộ

Một số ứng dụng hệ dãy

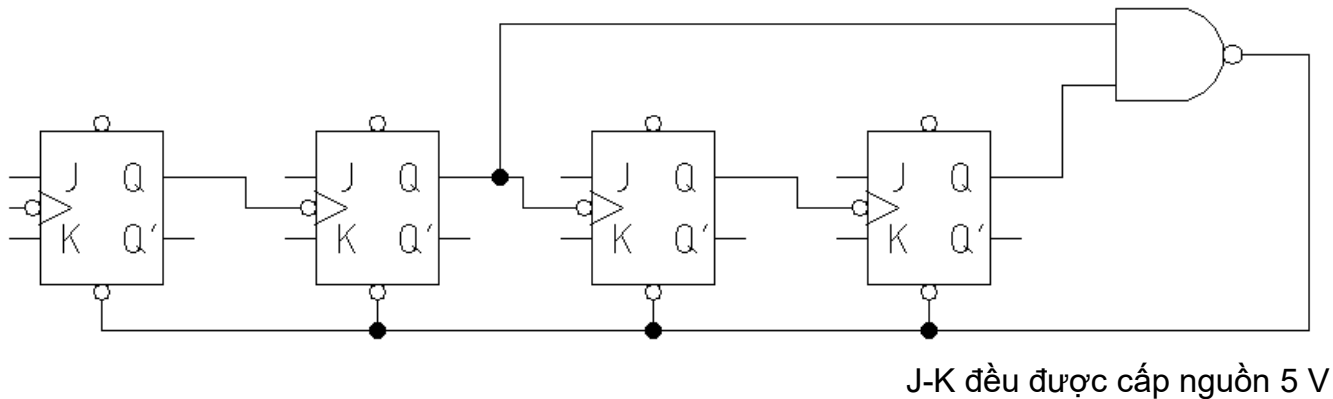
- Bộ đếm - đếm xung, bộ chia tần
 - bộ đếm module 16 không đồng bộ

Xung	q ₄	q ₃	q ₂	q ₁
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1
16	0	0	0	0



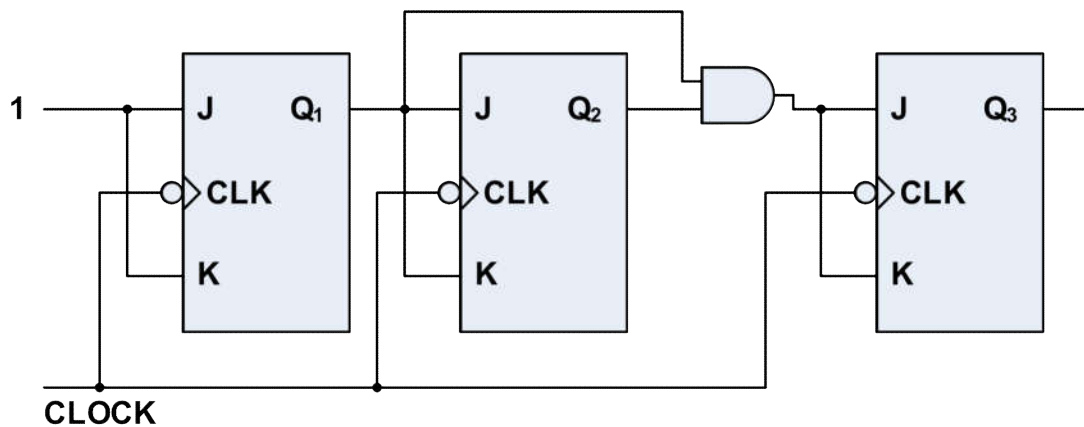
Một số ứng dụng hệ dây

- Bộ đếm - đếm xung, bộ chia tần
 - bộ đếm module 10 không đồng bộ
 - đếm đến xung 10 thì xoá về 0 bằng cách tạo xung tích cực vào chân CLR



Một số ứng dụng hệ dây

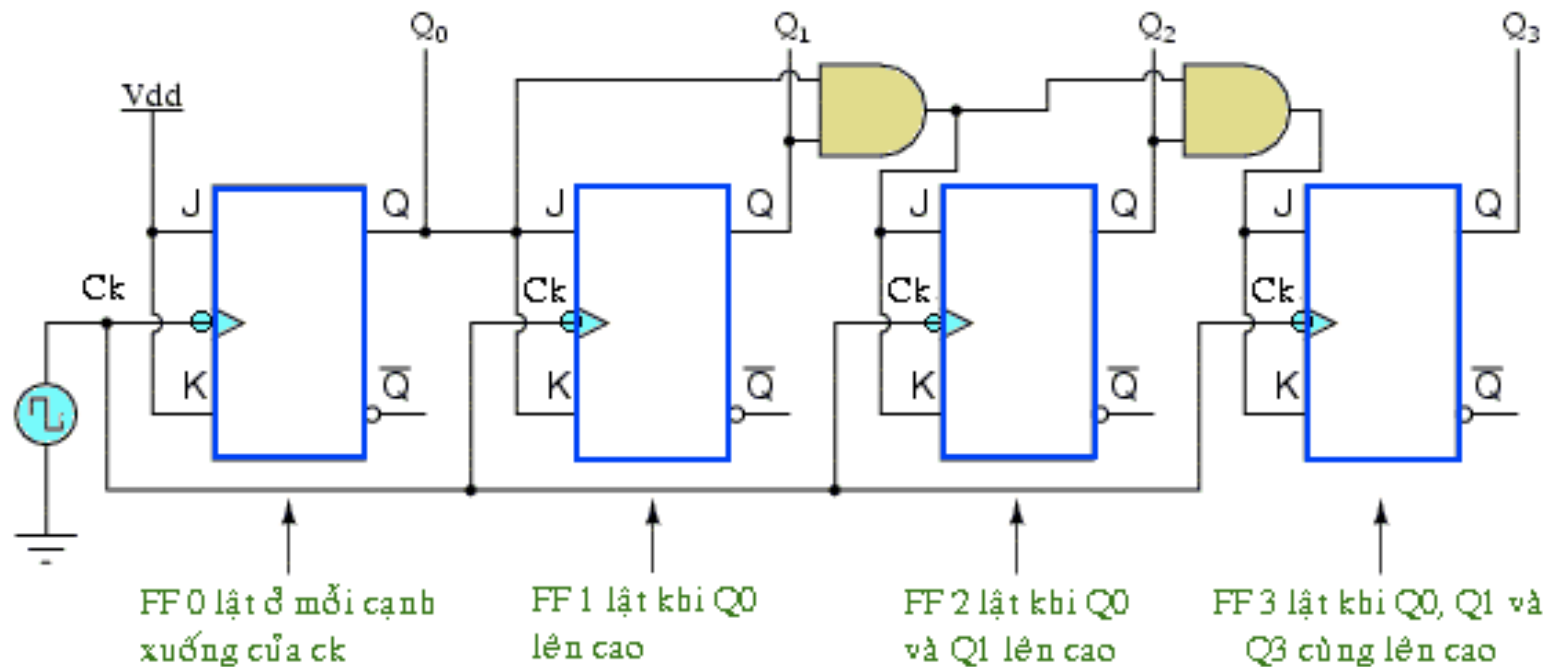
- Bộ đếm - đếm xung, bộ chia tần
 - bộ đếm module 8 đồng bộ



xung	q ₃	q ₂	q ₁
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1
8	0	0	0

Một số ứng dụng hệ dây

- Bộ đếm - đếm xung, bộ chia tần
 - bộ đếm module 16 đồng bộ



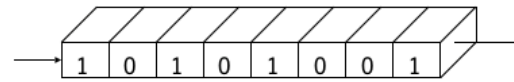
Một số ứng dụng hệ dây

- Thanh ghi
 - Có cấu tạo từ các trigger nối tiếp với nhau
 - Chức năng: lưu và dịch chuyển thông tin
 - Lưu ý: thanh ghi có thể được sử dụng làm bộ nhớ nhưng bộ nhớ không thể làm thanh ghi vì chỉ thanh ghi mới có chức năng dịch chuyển thông tin

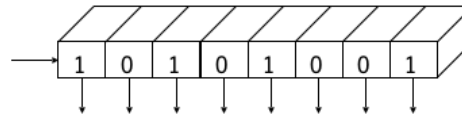
Một số ứng dụng hệ dây

- Thanh ghi

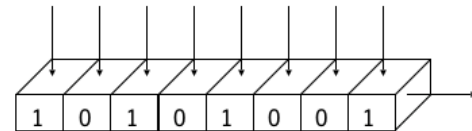
- vào nối tiếp - ra nối tiếp



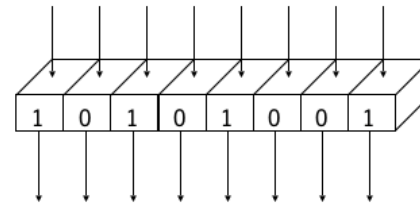
- vào nối tiếp - ra song song



- vào song song - ra nối tiếp

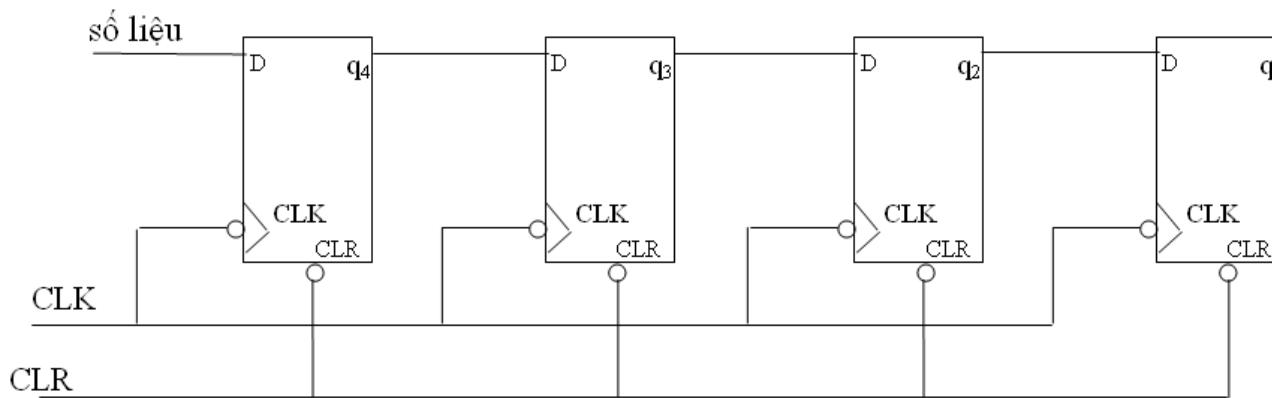


- vào song song - ra song song



Một số ứng dụng hệ dây

- Thanh ghi
 - Thanh ghi 4 bit vào nối tiếp - ra song song dùng D trigger



Dòng	Vào			Ra			
	CLR	số liệu	CLK	A	B	C	D
1	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0
3	1	1	1	1	0	0	0
4	1	1	2	1	1	0	0
5	1	1	3	1	1	1	0
6	1	0	4	0	1	1	1
7	1	0	5	0	0	1	1
8	1	0	6	0	0	0	1
9	1	0	7	0	0	0	0
10	1	0	8	0	0	0	0
11	1	1	9	1	0	0	0
12	1	0	10	0	1	0	0
13	1	0	11	0	0	1	0
14	1	0	12	0	0	0	1
15	1	0	13	0	0	0	0