

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA KỸ THUẬT MÁY TÍNH

TỔ CHỨC VÀ CẤU TRÚC MÁY TÍNH

II (IT012)



Sinh viên: Trần Nguyễn Thái Bình

MSSV: 23520161

Giảng viên hướng dẫn: Nguyễn Thành Nhân

Bài thực hành 1: IT012.O21.1 - Lab01

MỤC LỤC

1. Lý thuyết.....	3
2. Thực hành.....	3
2.1. MÔ PHỎNG CHỨC NĂNG CÁC CỔNG LUẬN LÝ	3
2.1.1. <i>NOT</i>	3
2.1.2. <i>AND</i>	3
2.1.3. <i>NAND</i>	3
2.1.4. <i>OR</i>	4
2.1.5. <i>NOR</i>	4
2.1.6. <i>XOR</i>	4
2.1.7. <i>XNOR</i>	6
2.2. MÔ PHỎNG CHỨC NĂNG CÁC THIẾT BỊ LƯU TRỮ'	8
2.2.1. <i>D-latch</i>	8
2.2.2. <i>D-Flipflop</i>	9
2.2.3. <i>Thanh ghi</i>	11
3. Bài tập.....	11
3.1. MÔ PHỎNG MẠCH TỔNG HỢP.....	11
3.2. MÔ PHỎNG MẠCH TỔ HỢP	13

1. Lý thuyết

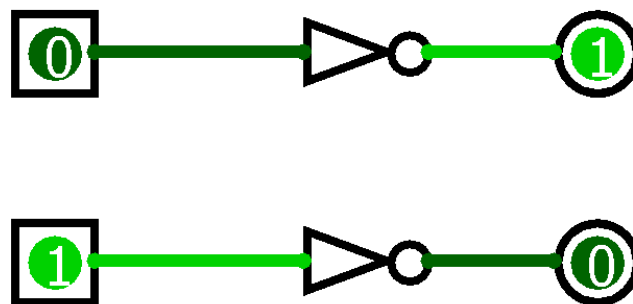
- Giảng viên hướng dẫn sinh viên sử dụng phần mềm Logisim dựa theo tài liệu: **Hướng dẫn sử dụng Logisim**

2. Thực hành

2.1. Mô phỏng chức năng các cổng luận lý

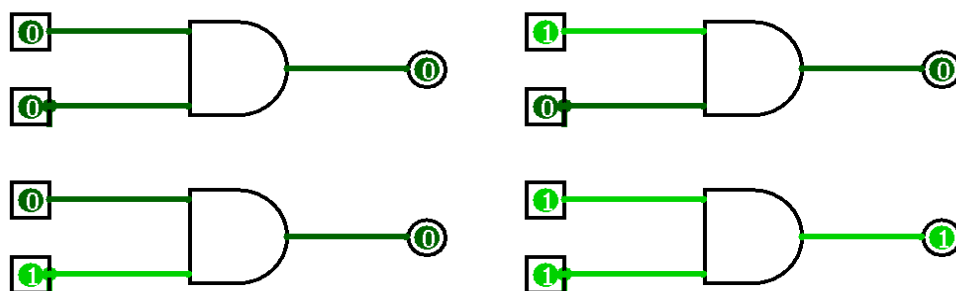
2.1.1. NOT

- Cổng NOT là cổng nhận một ngõ vào và một ngõ ra. Tín hiệu ngõ ra là sự phủ định của tín hiệu ngõ vào.



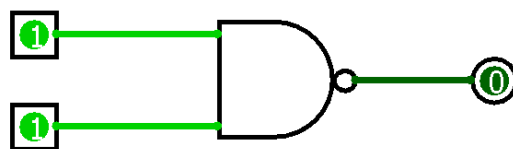
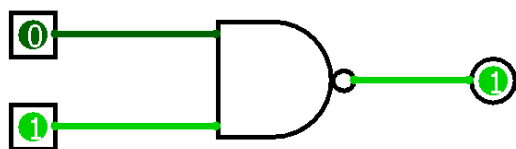
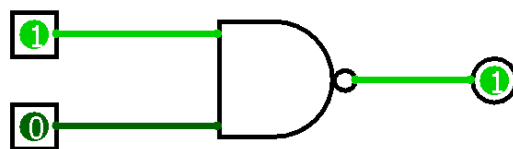
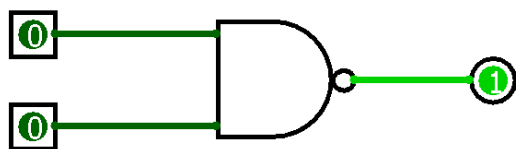
2.1.2. AND

- Cổng AND là cổng nhận nhiều ngõ vào và có một ngõ ra duy nhất. Nếu tất cả các ngõ vào đều có giá trị “1” thì ngõ ra cũng mang giá trị “1”. Ngược lại, nếu có ít nhất một ngõ vào nhận giá trị “0” thì ngõ ra mang giá trị “0”.



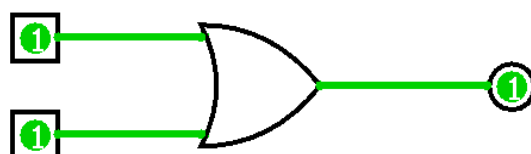
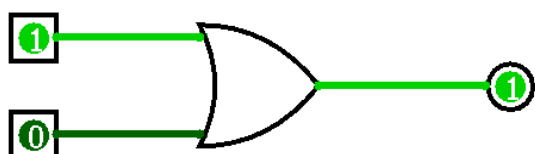
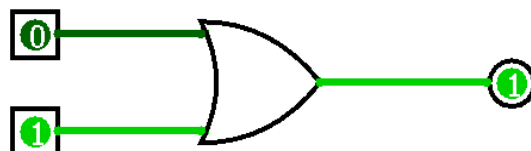
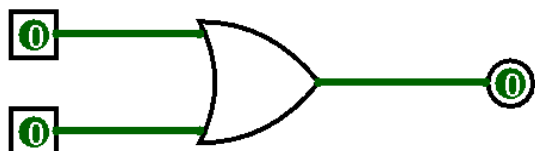
2.1.3. NAND

- “NAND” nghĩa là “NOT AND”, nghĩa là sự phủ định của cổng AND. Cổng NAND chỉ đưa tín hiệu “0” đến ngõ ra khi và chỉ khi nhận được tất cả các tín hiệu ngõ vào là “1”.



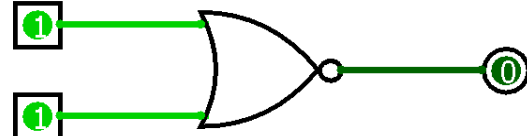
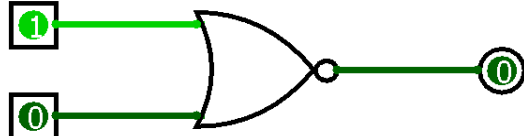
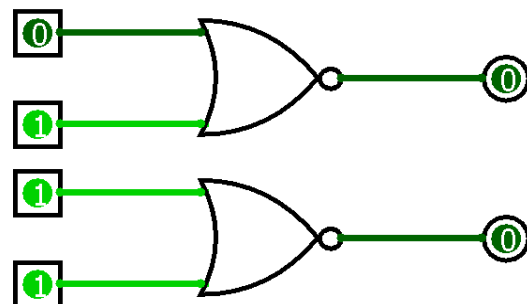
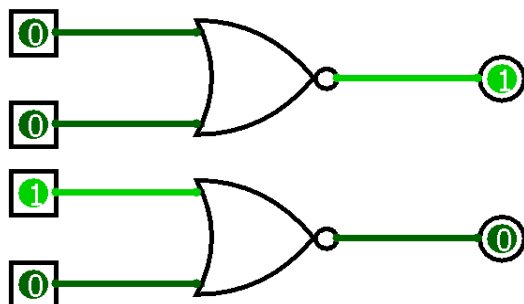
2.1.4. OR

- Cổng OR là cổng nhận nhiều ngõ vào và có một ngõ ra duy nhất. Nếu tất cả các ngõ vào đều có giá trị “0” thì ngõ ra cũng mang giá trị “0”. Ngược lại, nếu có ít nhất một ngõ vào nhận giá trị “1” thì ngõ ra mang giá trị “1”. Dưới đây là hình ảnh minh họa cho cổng OR có hai ngõ vào.



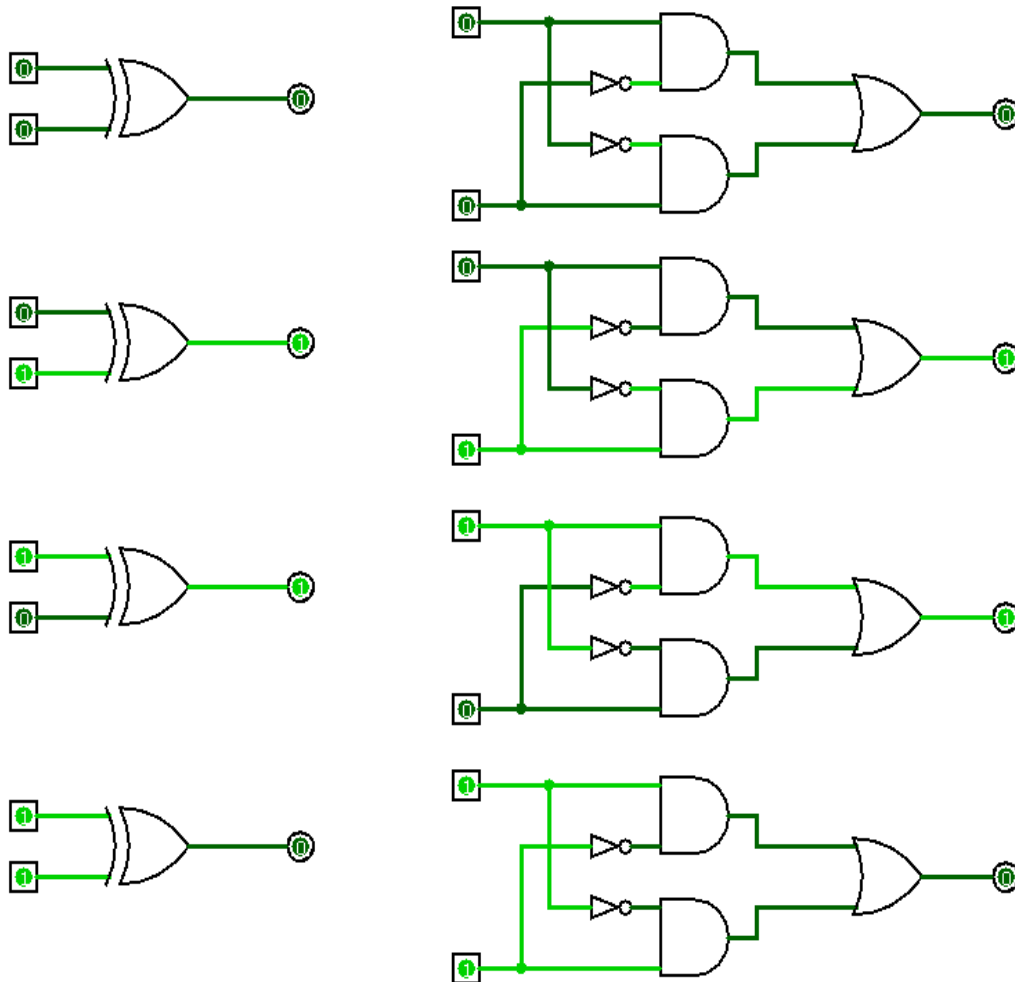
2.1.5. NOR

- “NOR” nghĩa là “NOT OR”, nghĩa là sự phủ định của cổng OR. Cổng NOR chỉ đưa tín hiệu “1” đến ngõ ra khi và chỉ khi nhận được tất cả các tín hiệu ngõ vào là “0”.



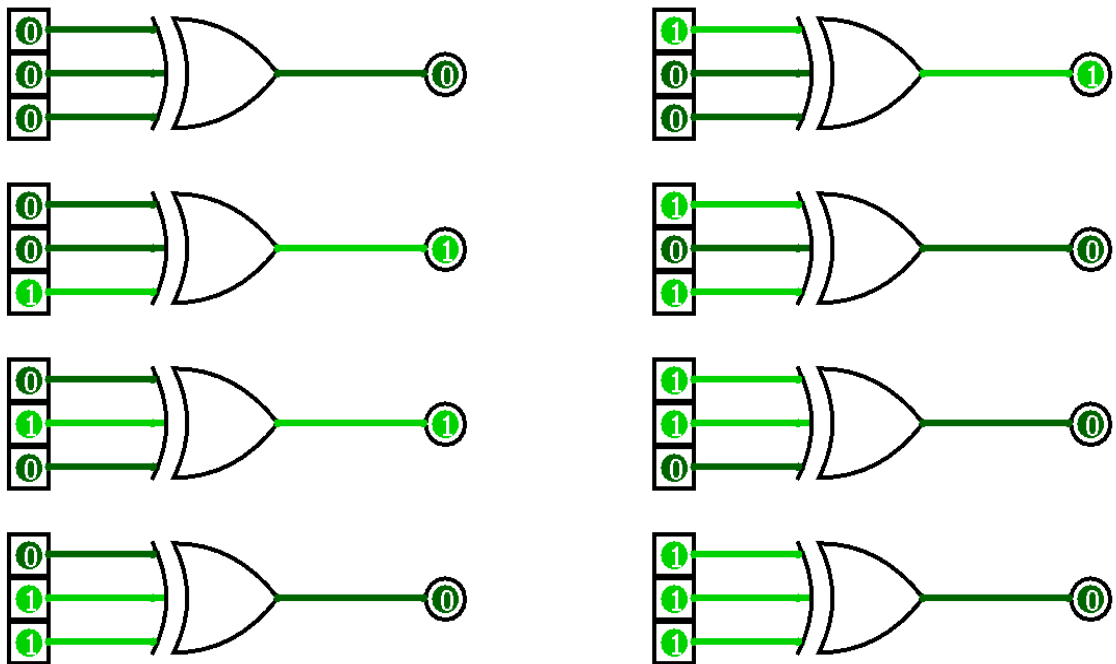
2.1.6. XOR

- XOR là cổng kết hợp của các cổng NOT, AND và OR. Cổng XOR nhận tín hiệu nhiều ngõ vào và có một ngõ ra.
- Đối với cổng XOR hai tín hiệu ngõ vào, kết quả trả về là “1” khi hai tín hiệu ngõ vào khác nhau. Ngược lại, khi hai tín hiệu ngõ vào giống nhau, tín hiệu ngõ ra là “0”.



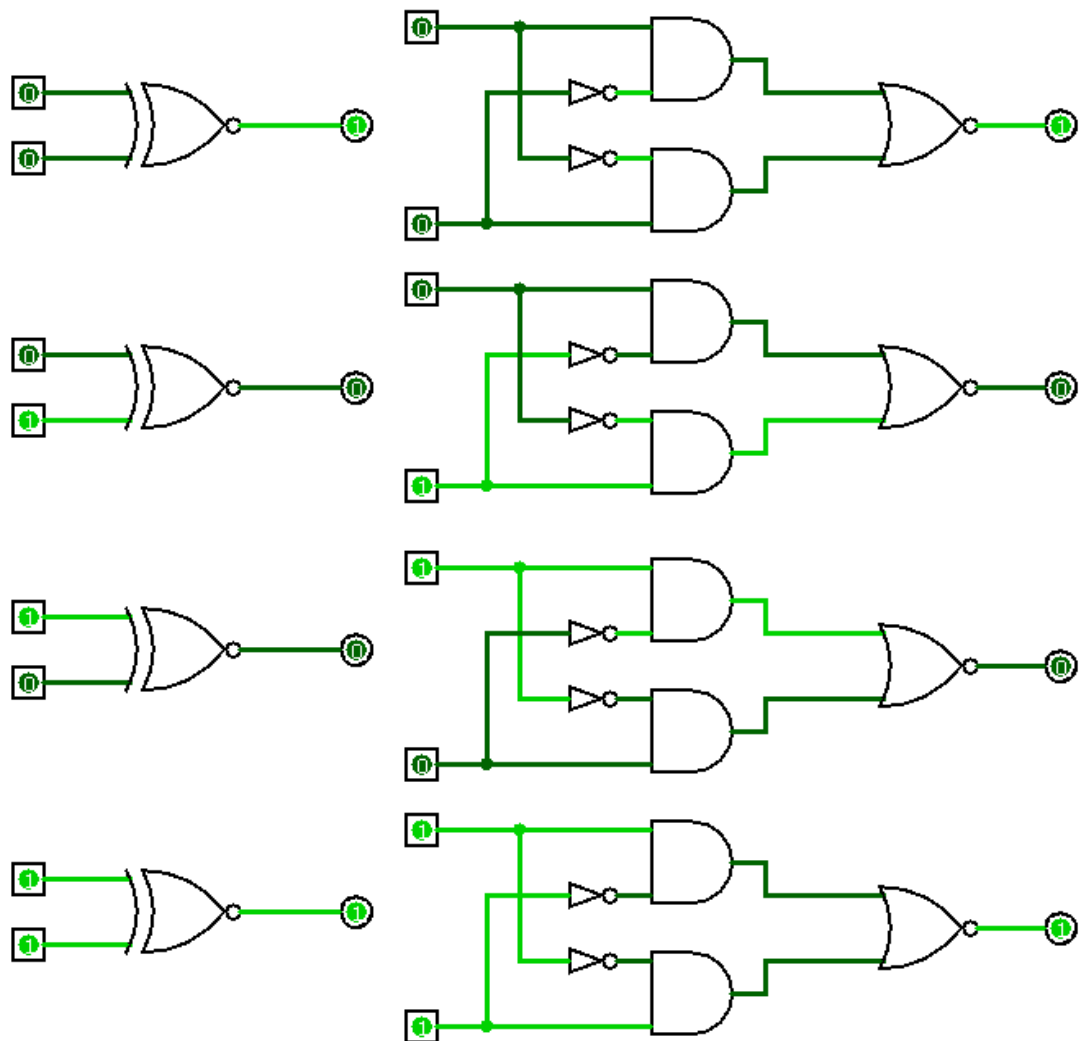
(bên trái là cổng XOR, bên phải là mô phỏng cổng XOR bằng các cổng NOT, AND và OR)

- Đối với cổng XOR có nhiều ngõ vào hơn, cổng chỉ trả về tín hiệu “1” khi và chỉ khi có duy nhất một tín hiệu ngõ vào “1”. Nếu có không hoặc nhiều hơn một tín hiệu ngõ vào “1”, tín hiệu ngõ ra sẽ là “0”.

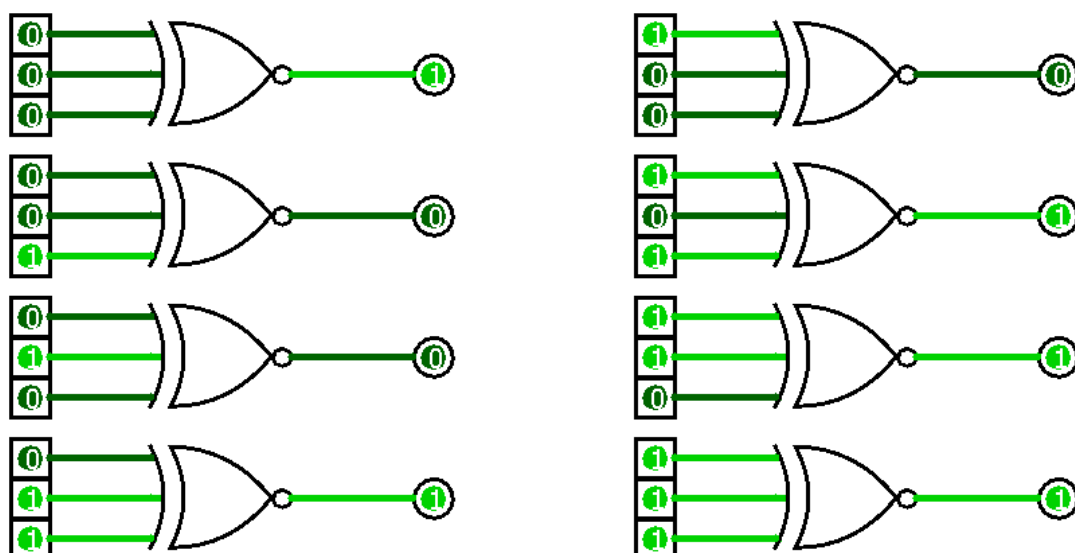


2.1.7. XNOR

- “XNOR” là “NOT XOR”, nghĩa là sự phủ định của cổng XOR. Cũng như cổng XOR, XNOR là cổng tổ hợp của NOT, AND và OR.
- Đối với XNOR hai ngõ vào, tín hiệu ngõ ra là “0” khi có hai tín hiệu ngõ vào khác nhau. Ngược lại, tín hiệu ngõ ra là “1”.



- Đối với XNOR nhiều hơn hai ngõ vào, tín hiệu ngõ ra sẽ ngược với cổng XOR nhiều ngõ. Đối với XNOR nhiều hơn hai ngõ vào, tín hiệu ngõ ra sẽ ngược với cổng XOR nhiều ngõ vào.



2.2. Mô phỏng chức năng các thiết bị lưu trữ

- Clock



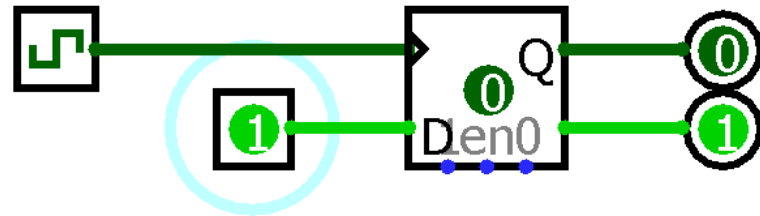
- Xung tín hiệu clock được truyền vào ngõ CLK là xung chữ nhật như trên hình vẽ. Mỗi chu kì gồm có 4 xung. Thứ tự đọc xung là thứ tự theo chiều mũi tên màu đen. Đối với cạnh ngang ở dưới (màu đỏ) là xung mức thấp (Low level), cạnh ngang ở trên (màu xanh lá) là xung mức cao (High level). Cạnh dọc đi lên từ mức Low \rightarrow High (màu tím) là cạnh lên (Rising edge). Cạnh dọc đi xuống từ mức High \rightarrow Low là cạnh xuống (Falling edge).

2.2.1. D-latch

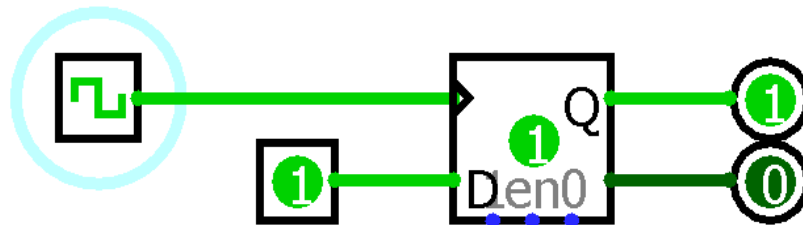
- Trong máy tính, dữ liệu được đưa vào theo các xung. Máy tính không thể ghi dữ liệu liên tục mà chỉ có thể ghi dữ liệu theo chu kì. D-Latch là thiết bị lưu trữ chỉ ghi nhận dữ liệu tích cực theo mức cao hoặc thấp, tùy theo cài đặt trên nó. Nhìn chung, D-Latch có các ngõ: E (Clock, nhận xung), ngõ vào D, ngõ ra Q (trước) và Q+ (sau).

E	Q+
0	Q
1	D

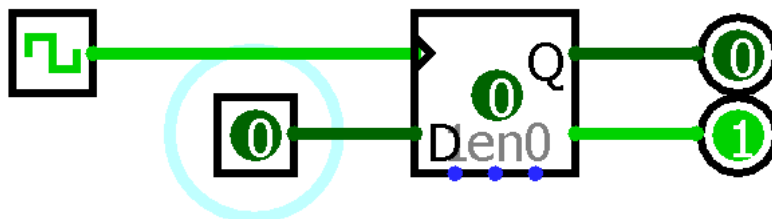
- Dưới đây là mô phỏng hoạt động của D-Latch tích cực mức cao (D Flip-Flop, trigger = High level, có 2 output, output thứ nhất chính là kết quả phụ thuộc đầu vào, output thứ 2 là NOT của kết quả)
 - 1. CLK: Low Level. Input = 1. Output = 0. Giả sử hiện tại ngõ E đang nhận xung ở mức thấp, nên dữ liệu dù được đưa vào D-Latch cũng sẽ không được ghi lại.



- 2. CLK: High Level. Input = 1. Output = 1. Tiếp theo ngõ E nhận xung ở mức cao, D-Latch ghi nhận dữ liệu. Ngõ ra trả về kết quả giống với dữ liệu đầu vào.



- 3. CLK: High Level. Input = 0. Output = 0. Trong thời gian ngõ E nhận xung ở mức cao, kết quả đầu ra sẽ luôn giống với dữ liệu đầu vào.



- Bảng chân trị cho ví dụ minh họa trên:

STEP	INPUT	CLK	OUTPUT
1	1	0	0
2	1	1	1
3	0	1	0

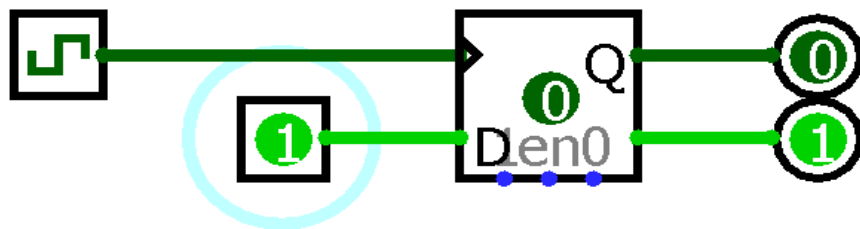
2.2.2. D-Flipflop

- Tương tự với D-Latch, D-Flipflop cũng là thiết bị lưu trữ dữ liệu theo xung. Cũng có các ngõ: CLK (Clock, nhận xung), ngõ vào D, ngõ ra Q (trước) và Q+ (sau). Nhưng thay vì tích cực theo mức, DFlipflop tích cực theo cạnh lên hoặc cạnh xuống.

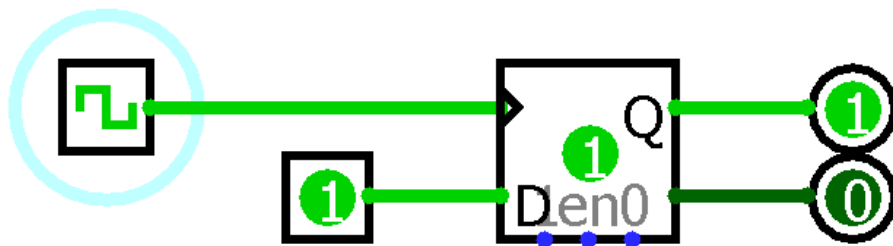
CLK	Q+
-	Q
↑	D

- Dưới đây là mô phỏng hoạt động của D-Flipflop tích cực cạnh lên (D Flip-Flop, trigger = Rising Edge, có 2 output, output thứ nhất chính là kết quả phụ thuộc đầu vào, output thứ 2 là NOT của kết quả)

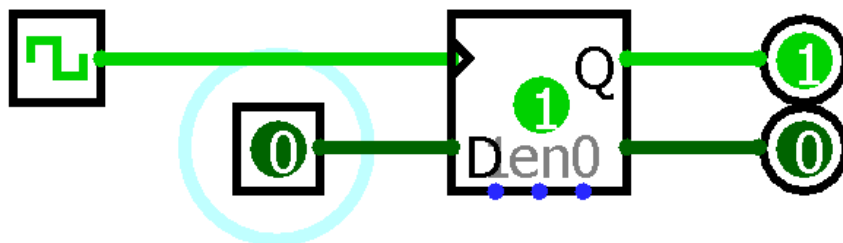
- 1. CLK: Low Level. Input = 1. Output = 0. Mức xung đang thấp, không có cạnh lên nên kết quả đầu ra không thay đổi theo dữ liệu đầu vào.



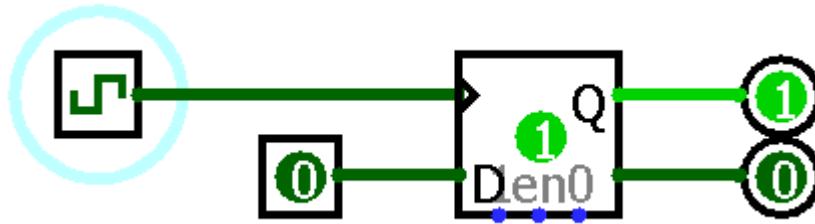
- 2. CLK: High Level. Input = 1. Output = 1. Thay đổi xung lên mức cao, xuất hiện cạnh lên, lúc này kết quả đầu ra thay đổi giống như dữ liệu đầu vào.



- 3. CLK: High Level. Input = 0. Output = 1.



- 4. CLK: Low Level. Input = 0. Output = 1. Thay đổi xung xuống mức thấp, không xuất hiện cạnh lên, kết quả không thay đổi.



- Bảng chân trị cho ví dụ mô phỏng trên:

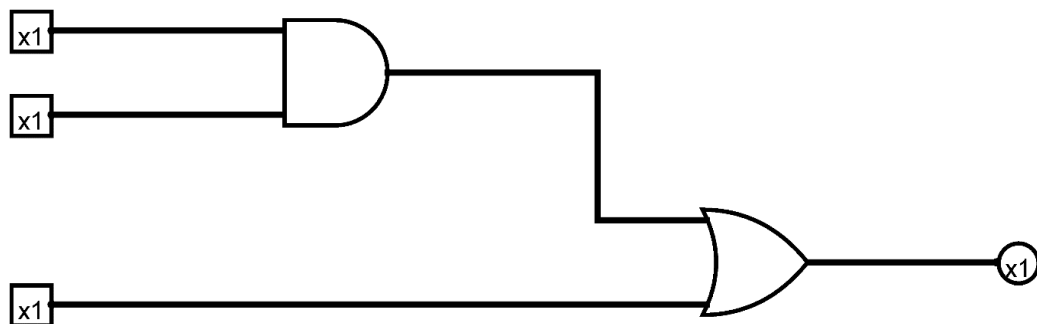
STEP	INPUT	CLK	OUTPUT
1	1	0	0
2	1	↑	1
	1	1	1
3	0	1	1
4	0	↓	1
	0	0	1

2.2.3. Thanh ghi

- Thanh ghi là sự kết hợp của nhiều D-Flipflop vào một thiết bị duy nhất, dùng chung một ngõ CLK. Tương tự với D-Flipflop, thanh ghi cũng có ngõ vào D, ngõ ra Q và ngõ CLK (Enable/E). Khác với D-Flipflop chỉ lưu trữ được 1-bit dữ liệu, thanh ghi lưu trữ được nhiều hơn thế.
- Mô phỏng tại phần 3.2.

3. Bài tập

3.1. Mô phỏng mạch tổng hợp



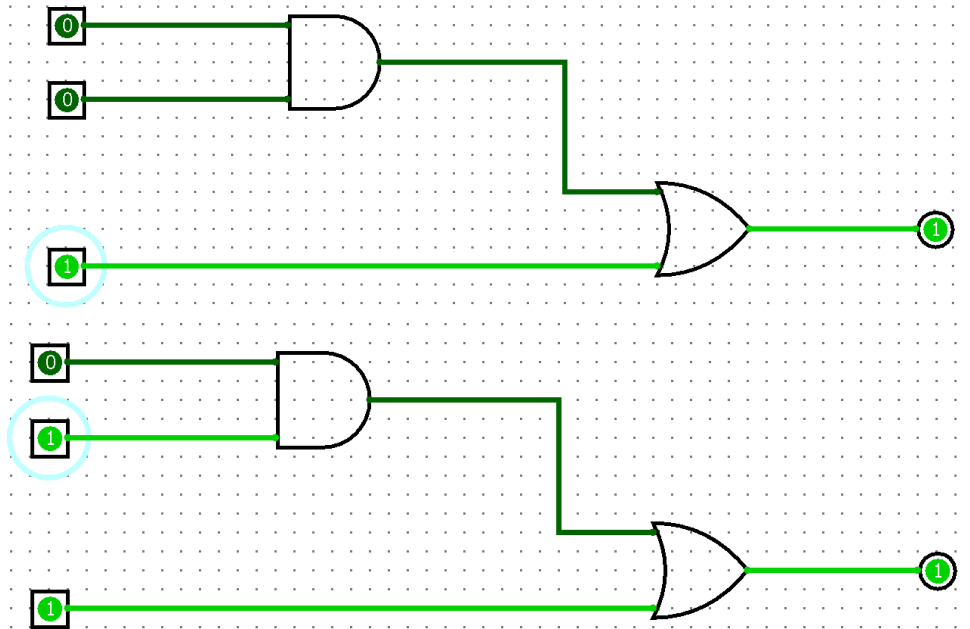
- Mạch tổ hợp này gồm có 1 cổng AND, 1 cổng OR, 3 ngõ vào, 1 ngõ ra trung gian và 1 ngõ ra chính. Hoạt động của mạch này được minh họa bởi bảng chân trị sau (A, B, C là các ngõ vào của mạch theo thứ tự từ trên xuống dưới, D là ngõ ra):

A	B	C	D
x	x	1	1
1	1	x	1

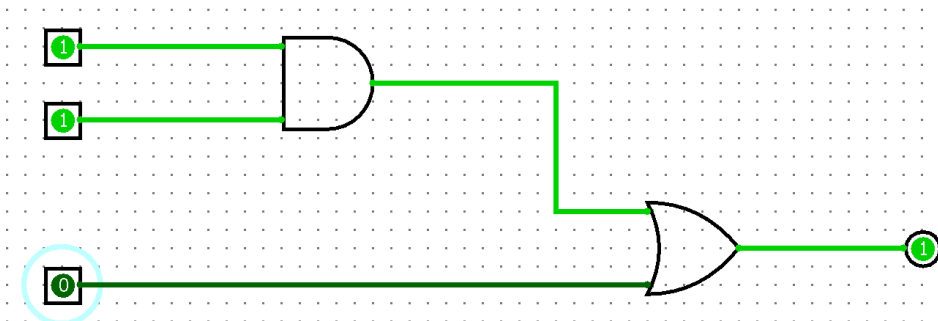
1	0	0	0
0	1	0	0

- Trong đó:

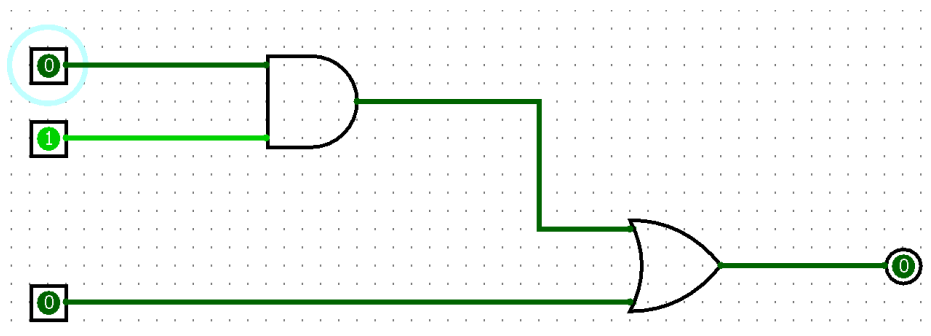
- **Dòng 1:** $C = 1, D = 1$. Vì C đi qua cổng OR nên bất kể A, B là giá trị gì.

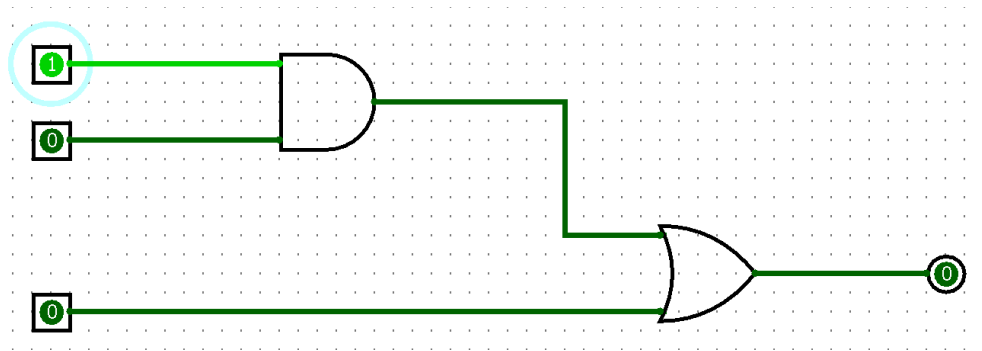


- **Dòng 2:** $A = B = 1, D = 1$. Vì A và B cùng đi qua cổng AND nên giá trị ra tức giá trị vào cổng OR đã nhận 1, cho nên $D = 1$.



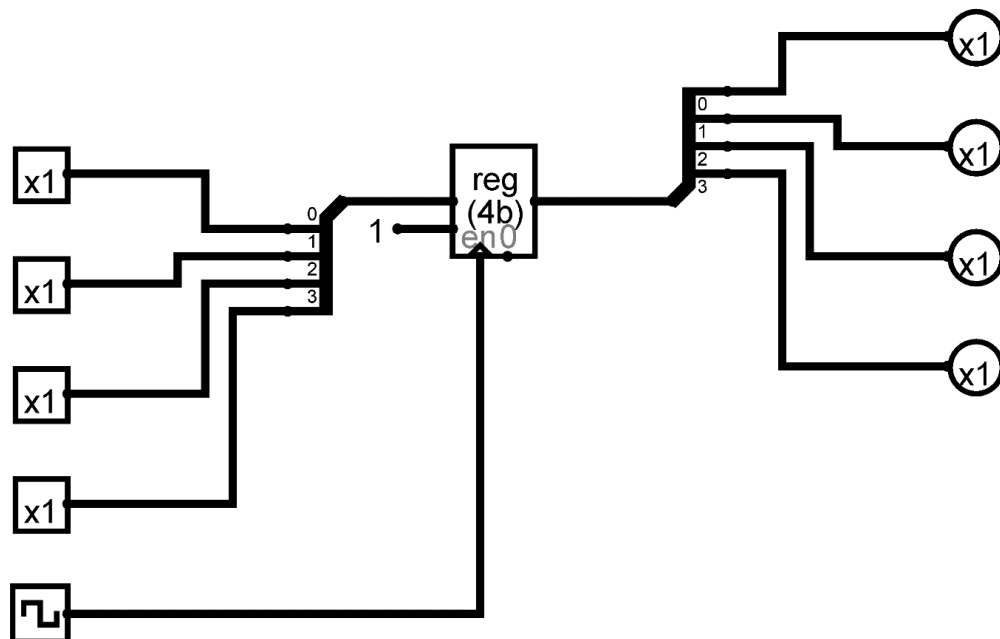
- **Dòng 3, 4:** A hoặc B có giá trị 0, đầu ra của cổng AND sẽ có giá trị 0, kết hợp với $C = 0$ nên $D = 0$.



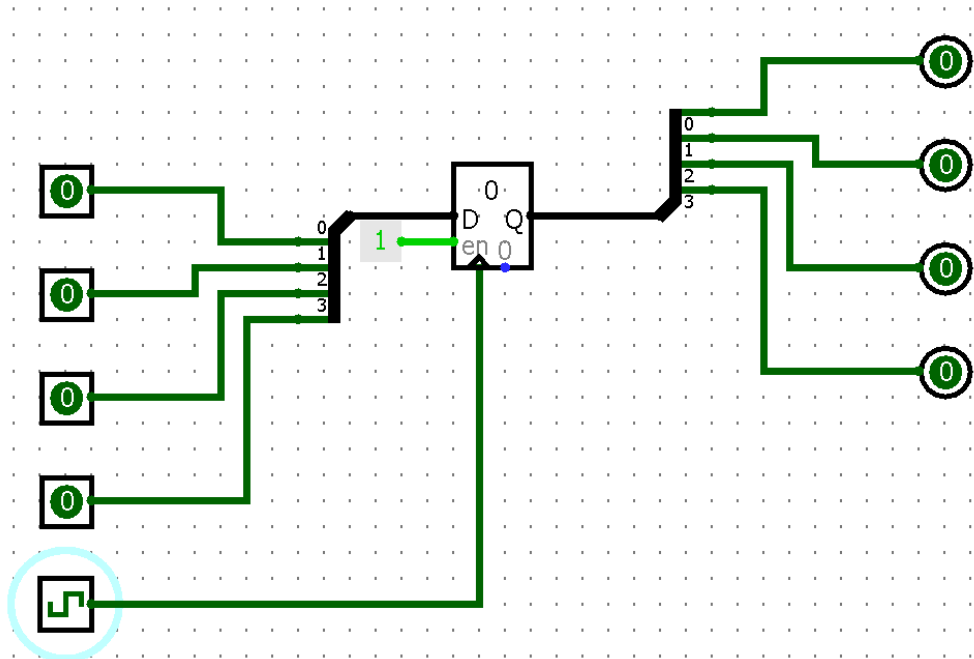


3.2. Mô phỏng mạch tổ hợp

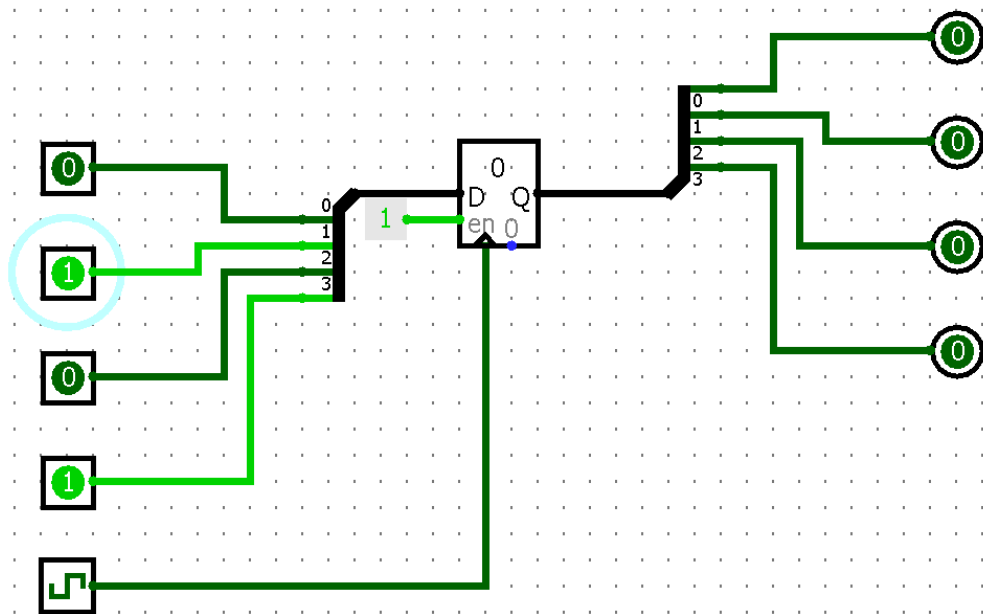
- Bộ tách bus và gộp bus có các thuộc tính **Fan Out: 4** và **Bit Width In: 4**
- Thanh ghi có thuộc tính **Data Bits: 4**



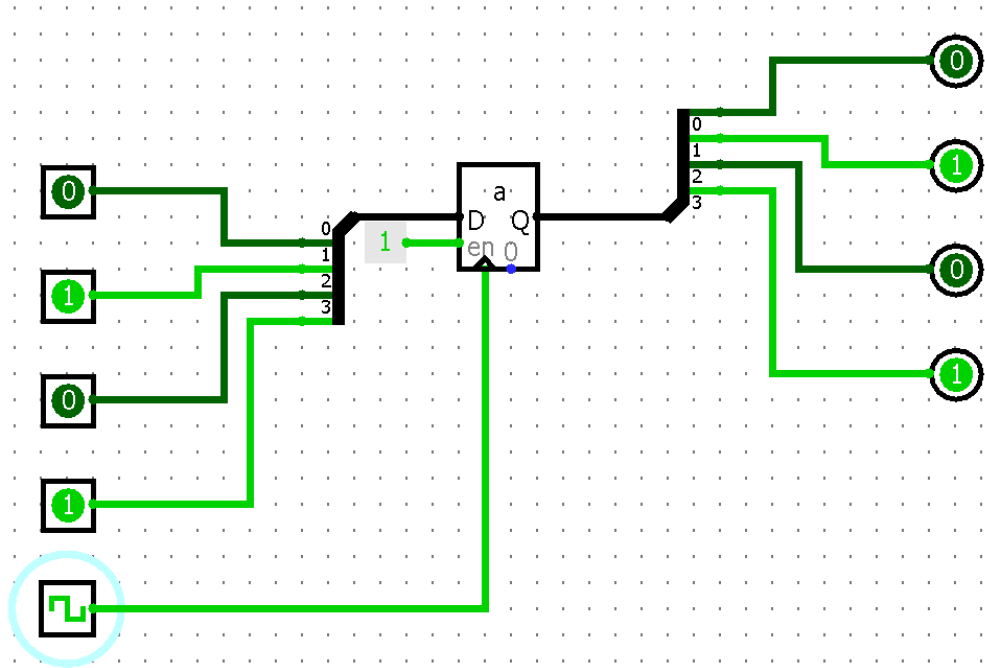
- Đây là mạch của thanh ghi Register: Kết hợp bởi nhiều D-Flipflop.
- Register Trigger: Rising Edge.
- Output sẽ thay đổi theo input khi CLK ở Rising Edge.
- Ngược lại, output sẽ giữ nguyên khi CLK ở Falling Edge, Low Level, High Level.
- Ví dụ minh họa:
 - o 1. CLK: Low Level.



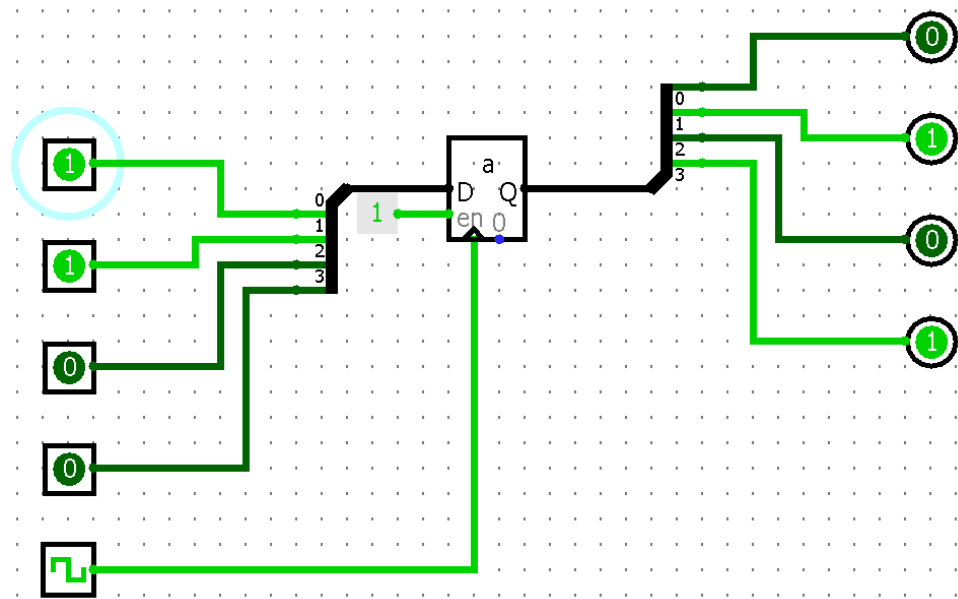
- 2. Thay đổi input, output không thay đổi.



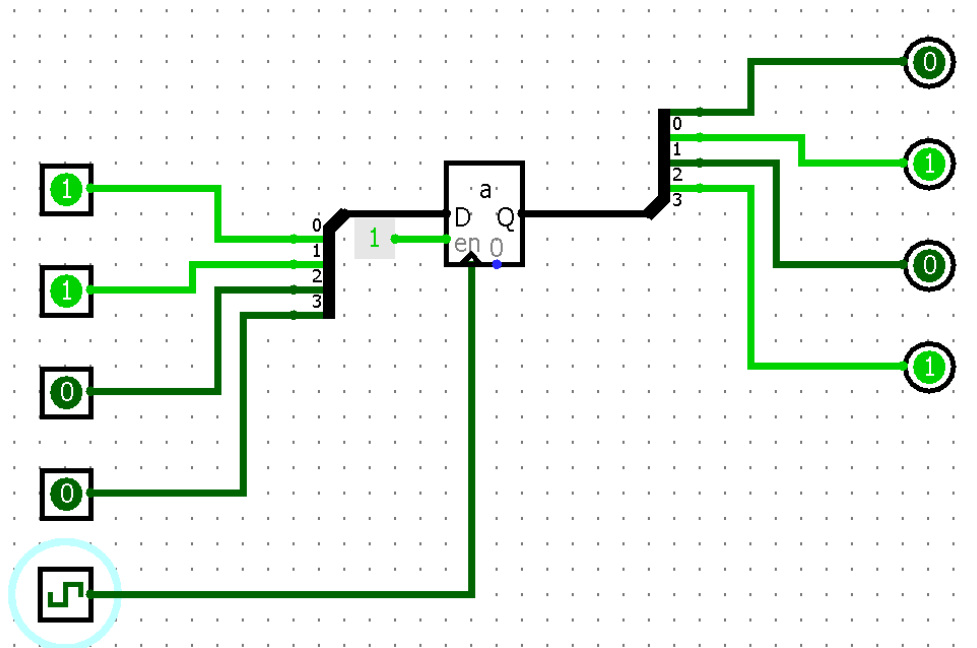
- 3. CLK: High Level. Tức vừa trải qua giai đoạn Rising Edge. Output đã được thay đổi.



- 4. Thay đổi input, output không thay đổi.



- 5. CLK: Low Level. Tức vừa trải qua giai đoạn Falling Edge. Output vẫn không thay đổi.



- Bảng chân trị cho ví dụ trên:

STEP	INPUT	CLK	OUTPUT
1	0000	0	0000
2	0101	0	0000
3	0101	↑	0101
	0101	1	0101
4	1100	1	0101
5	1100	↓	0101
	1100	0	0101