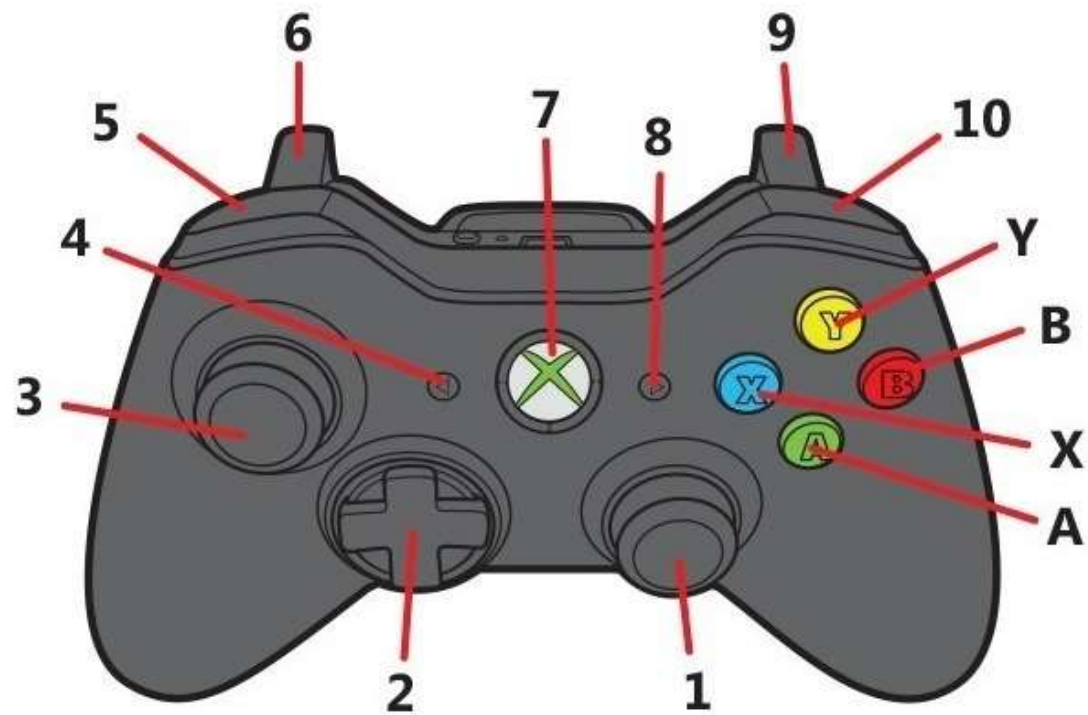


Codificadores



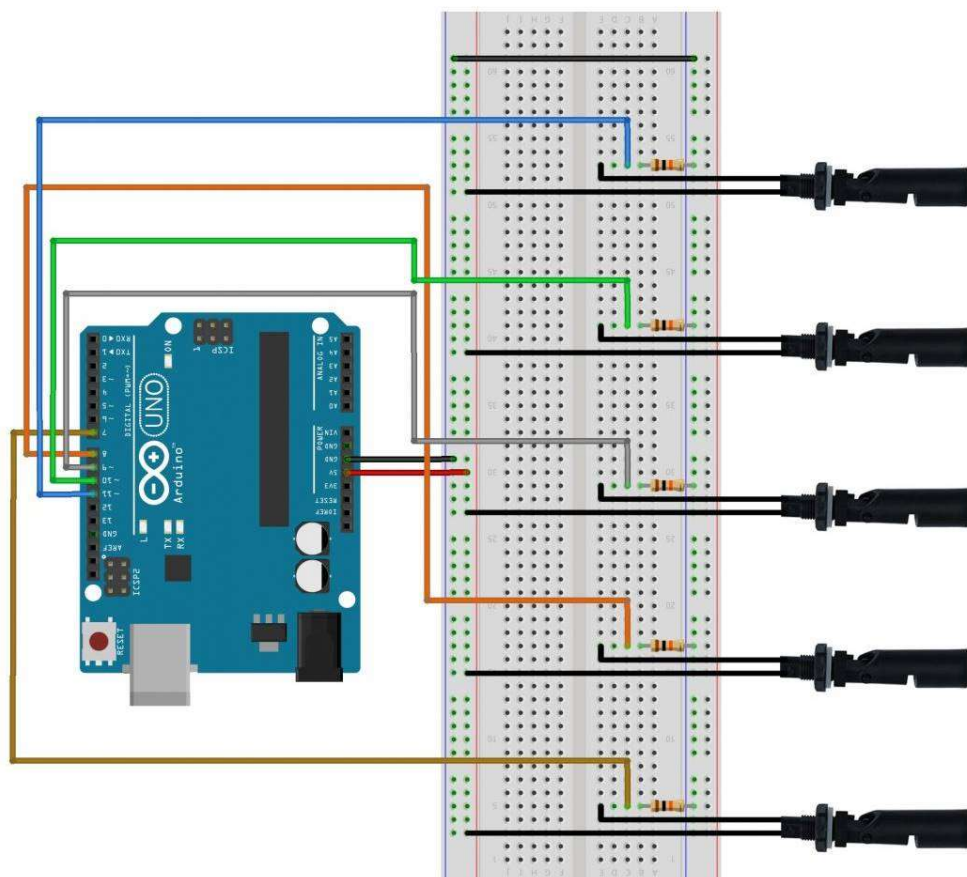
Aula 13



<https://br.ccm.net/faq/43629-como-resetar-seu-console-xbox-360>







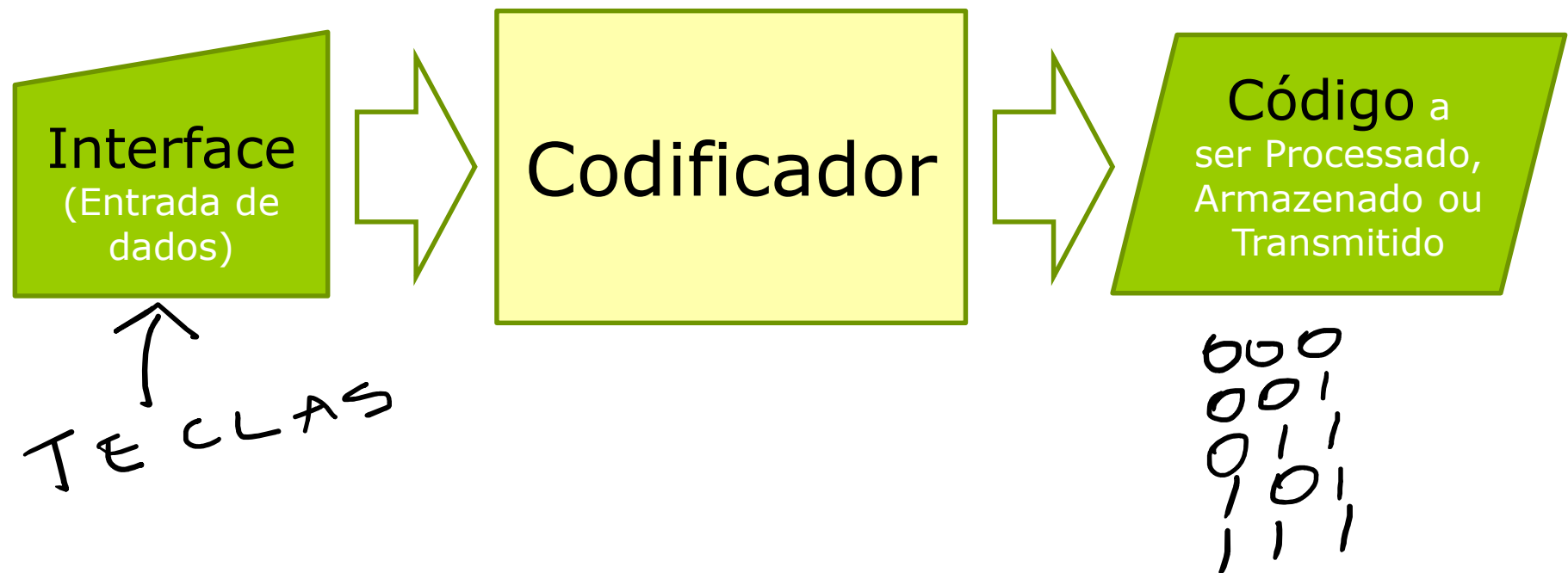
<https://www.usinainfo.com.br/blog/sensor-de-nivel-de-agua-com-arduino-automacao-residencial-de-controle/>

Códigos

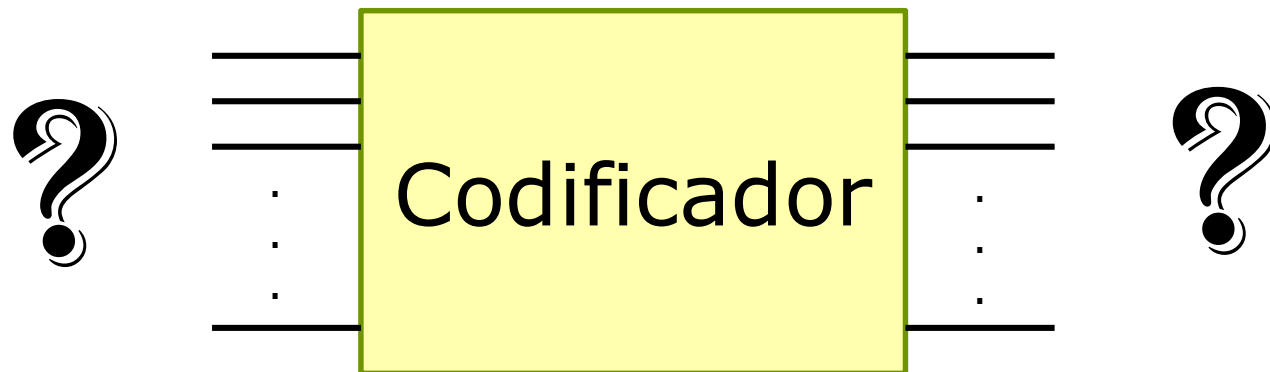
- São símbolos ou combinação deles, representados por números, letras, palavras, desenhos, imagens, sons etc que estabelecem uma determinada característica de comunicação entre dois sistemas.

Codificar significa transformar informações conhecidas, de uso comum e de fácil entendimento, em um conjunto de símbolos, letras, números ou palavras de forma a **minimizar ou facilitar** o armazenamento, o processamento e a transmissão da informação original. Em sistemas digitais, na maioria dos casos, codificar significa transformar um número decimal em um número binário para a manipulação nesses sistemas.

Como dimensionar e projetar um codificador



Quantas entradas e quantas saídas terá o codificador?



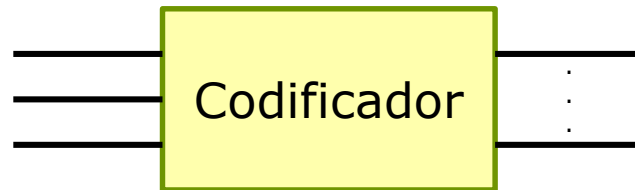
Esta é sempre a primeira pergunta a ser feita, pois vai definir o tamanho do projeto e a tecnologia que poderá ser utilizada.

Entradas

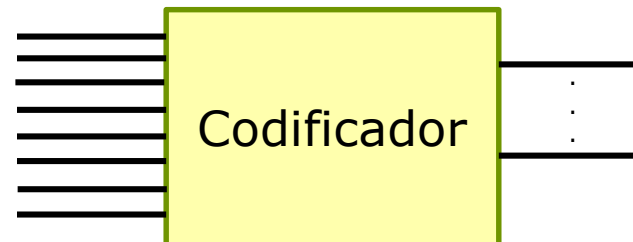
A quantidade de entradas depende do número de dados que serão codificados.
No caso de um sistema com teclas, quantas teclas tem dados que serão codificados;
No caso de uma rede de sensores, quantos sensores serão codificados etc.

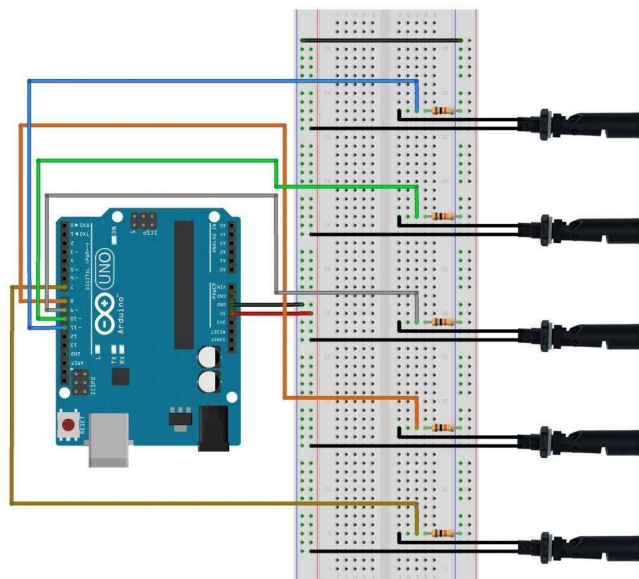
Exemplos:

3 botões a serem codificados = 3 entradas

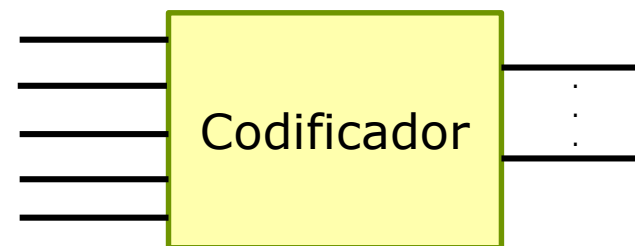


8 botões a serem codificados = 8 entradas





5 sensores a serem codificados = 5 entradas





107 teclas a serem codificados = 107 entradas

OBS: Quando temos uma grande quantidade de entradas, a tecnologia a ser usada na codificação é outra, que será visto em momento oportuno.

Saídas

- A quantidade de saídas depende da quantidade de informações que foram codificadas e de como foi feita a codificação.

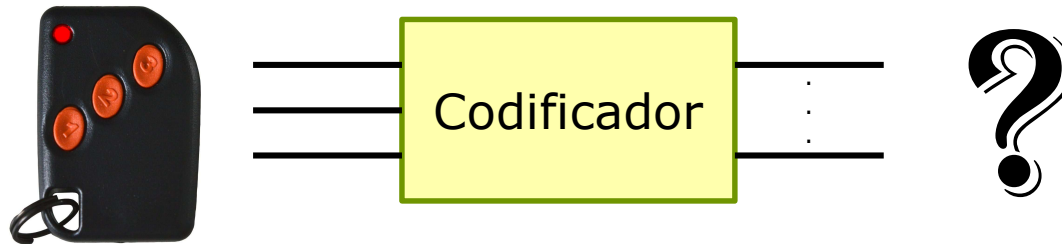
Se codificarmos um código de barras, sempre teremos 5 saídas, pois isto é o padrão do código, se codificarmos um ASC II de 128 símbolos, sempre teremos 7 saídas, mesmo se não usarmos todos os símbolos.

Mas veremos aqui com determinar o número de saídas pela quantidade de informações codificadas, para isso temos que saber como funcionará a interface de entrada para vermos as informações da saída, por exemplo:

- Apenas uma tecla é apertada por vez? Ou pode ter duas teclas apertadas juntas, gerando uma função diferente das funções de quando cada tecla é pressionada isoladamente?

Ou seja, temos que saber **quantas funções** teremos neste sistema.

Exemplos:



Funcionamento do sistema do controle de acionamento de um portão automatizado.

Pressionando o botão 1 o portão abre. Não fecha automaticamente.

Pressionando o botão 2 o portão fecha.

Pressionando o botão 3 o portão abre e após 10 segundos ele fecha automaticamente.

Nestes caso temos 3 funções diferentes a serem executadas, então podemos atribuir um número para cada uma destas funções, seguindo o número das teclas.

Por exemplo:

1 – portão abre

2 – portão fecha

3 – portão abre e fecha após os 10 segundos.

Mas a saída deve ser um número binário,
pois ali temos somente sinais com valor verdadeiro ou falso, 1 ou 0.

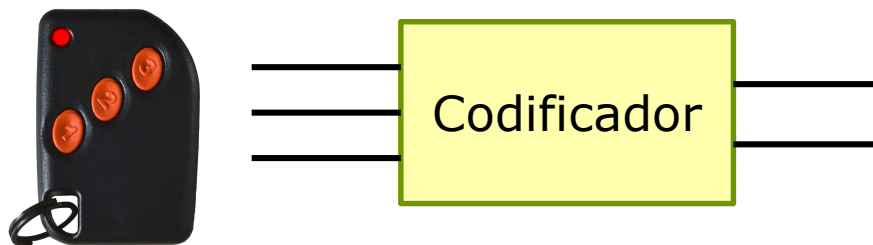
Então, o que temos na saída é:

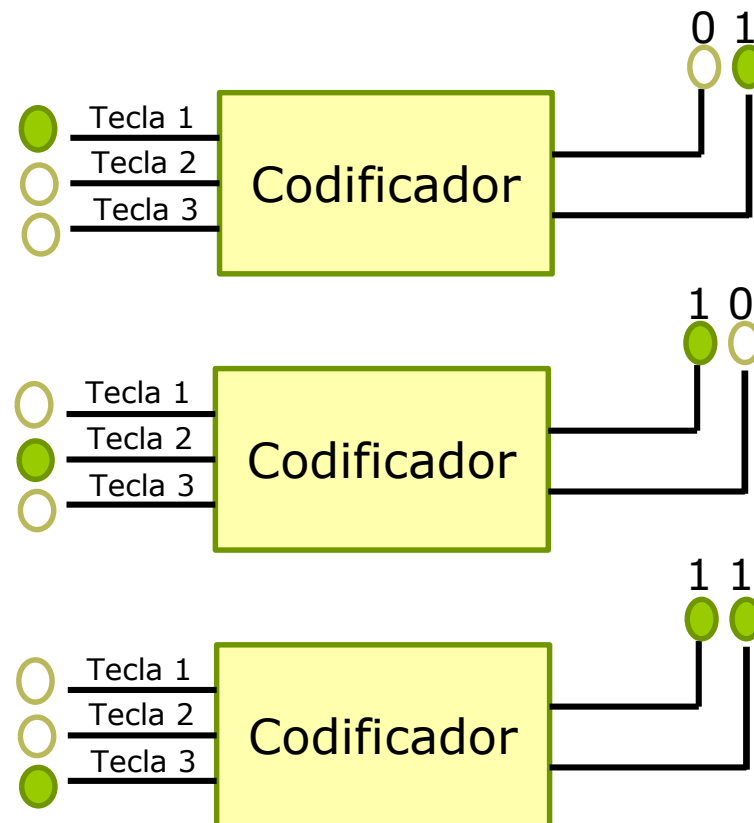
1 = 01

2 = 10

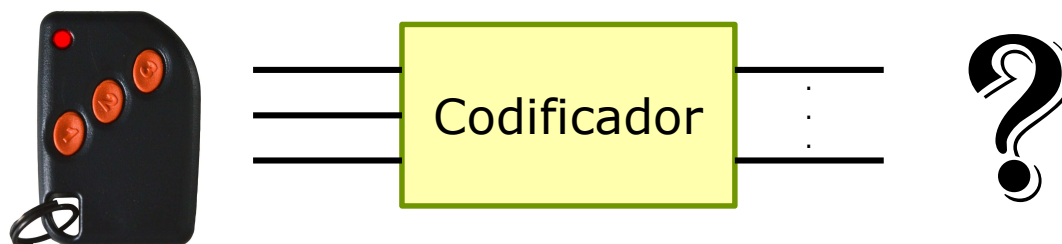
3 = 11

Ou seja, para este sistema temos 2 saída, uma para cada bit.





Outro exemplo:



Mas agora, este mesmo sistema pode ter mais funções:

Pressionando o botão 1 o portão abre. Não fecha automaticamente.

Pressionando o botão 2 o portão fecha.

Pressionando o botão 3 o portão abre e após 10 segundos ele fecha automaticamente.

Pressionando os botões 1 e 2 juntos, o portão abre com velocidade acelerada.

Pressionando os botões 1 e 3 juntos, o portão fecha com velocidade acelerada.

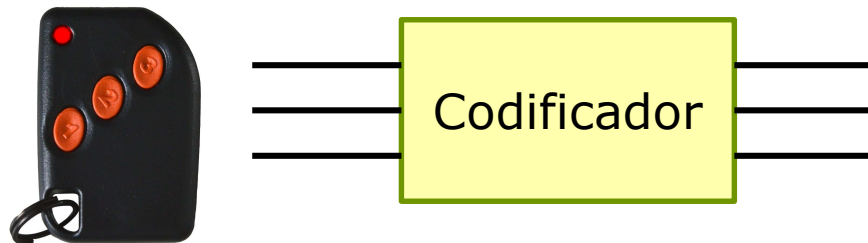
Ou seja, agora temos 5 funções e não conseguimos mais fazer usando apenas 2 bits. Agora temos que usar 3 bits pois a contagem irá até 5 se começarmos em 1 ou até 4 se começarmos em 0, o que não muda a quantidade de saídas, pois ambos os valores precisam no mínimo de 3 bits para serem representados.

Então fica:

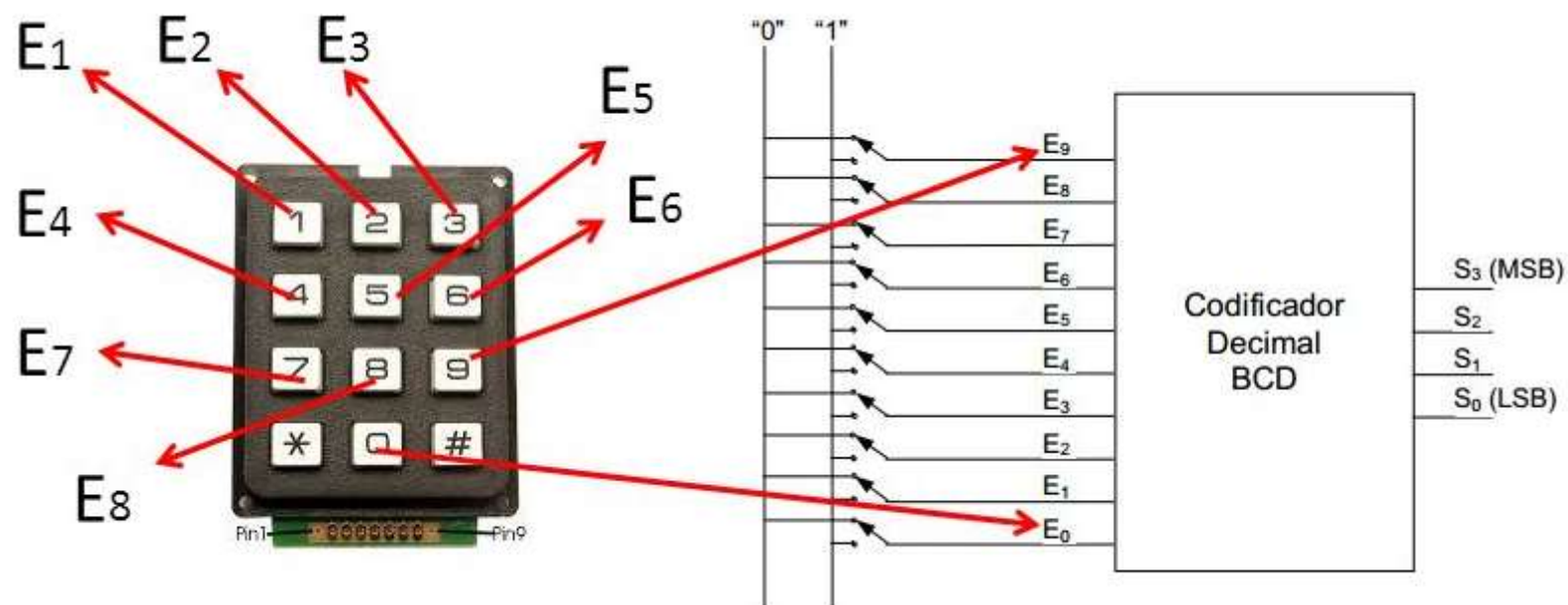
1 = 001
2 = 010
3 = 011
4 = 100
5 = 101

Temos aí 5 combinações diferentes dos valores de saída.
Lembre-se que com 3 bits de saída poderemos representar até 8 valores diferentes.

$N = 2^n$ onde N = número de combinações e n o número de saídas (número de bits).



E para um codificador Decimal para binário??

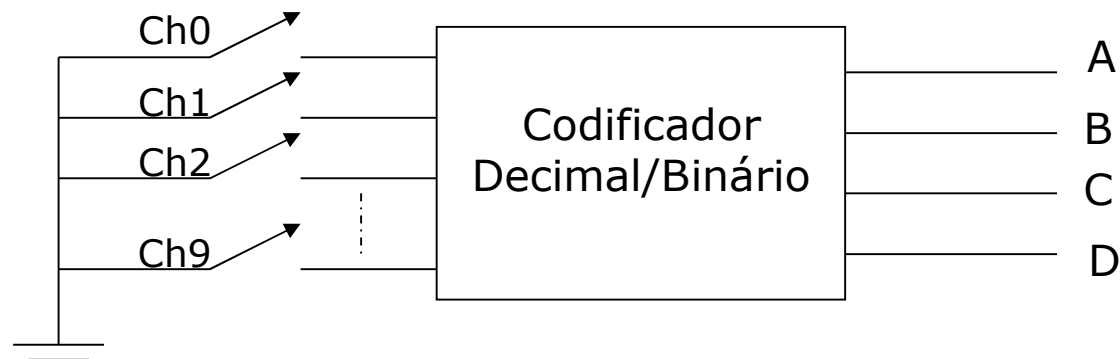


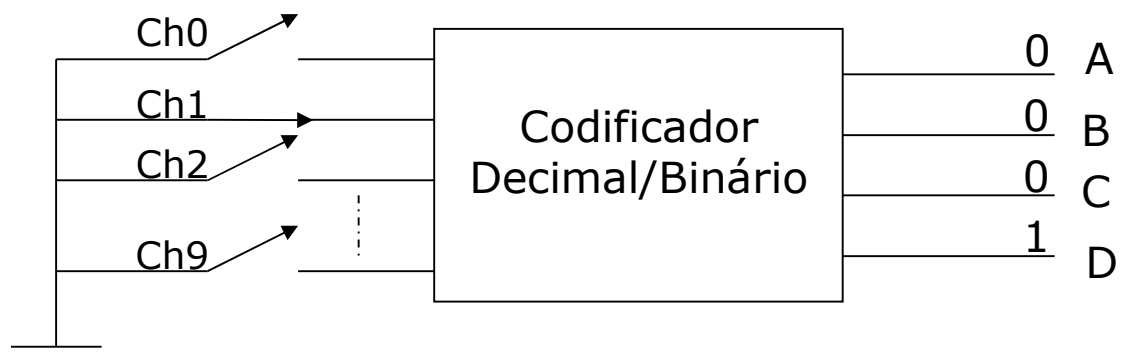
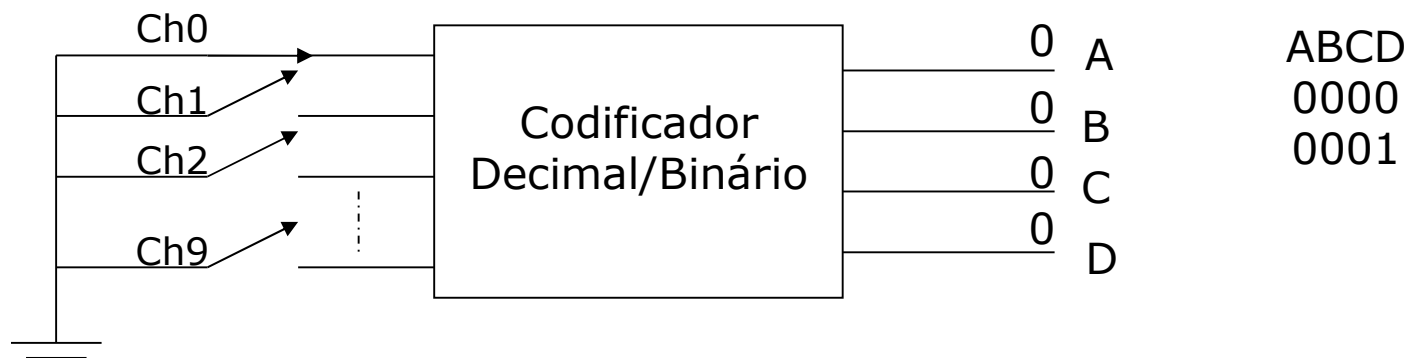
<https://slideplayer.com.br/slide/10339486/>

Codificador Decimal/Binário

A entrada do código decimal é feita através de um conjunto de chaves numeradas de 0 a 9 e a saída por 4 fios, para fornecer um código binário de 4 bits, correspondente à chave acionada.

Obs: A chave fechada equivale a nível lógico 0, para evitar o problema prático, principalmente da família TTL, do terminal aberto seja equivalente a nível lógico 1.





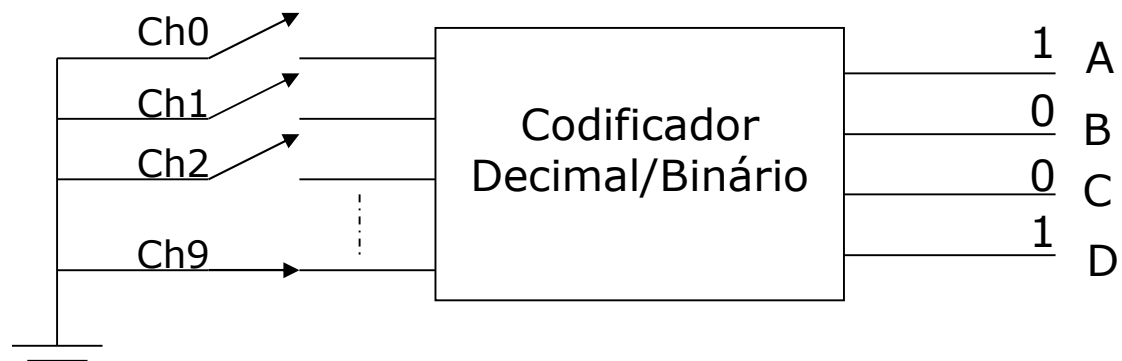
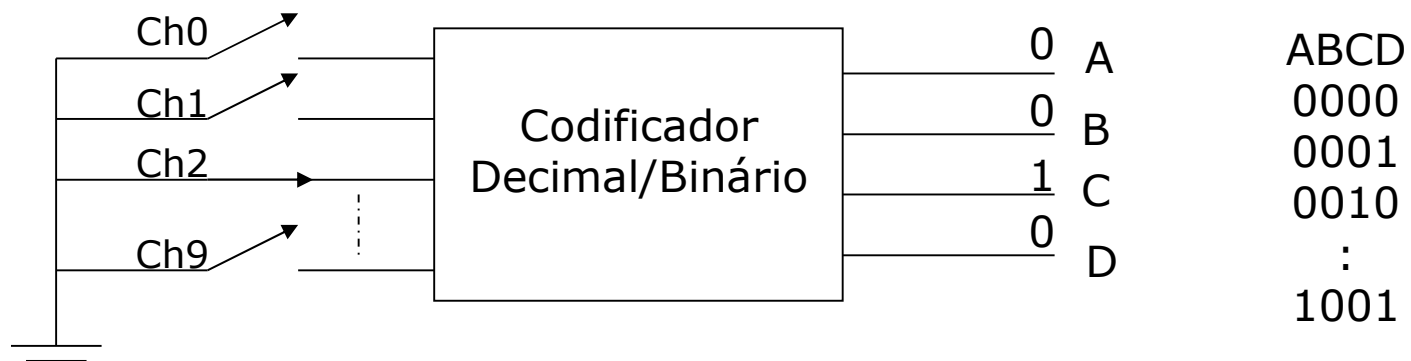
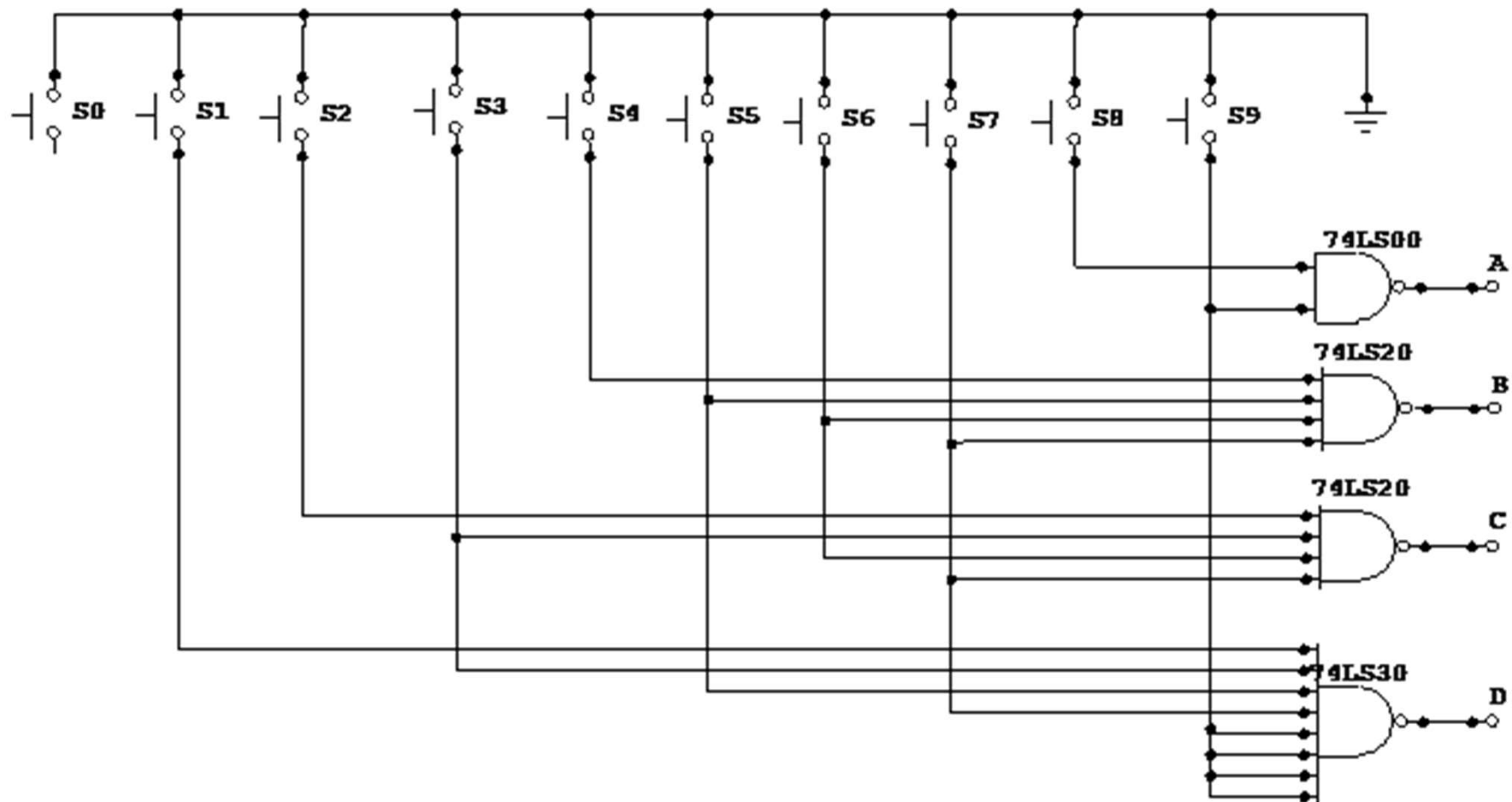


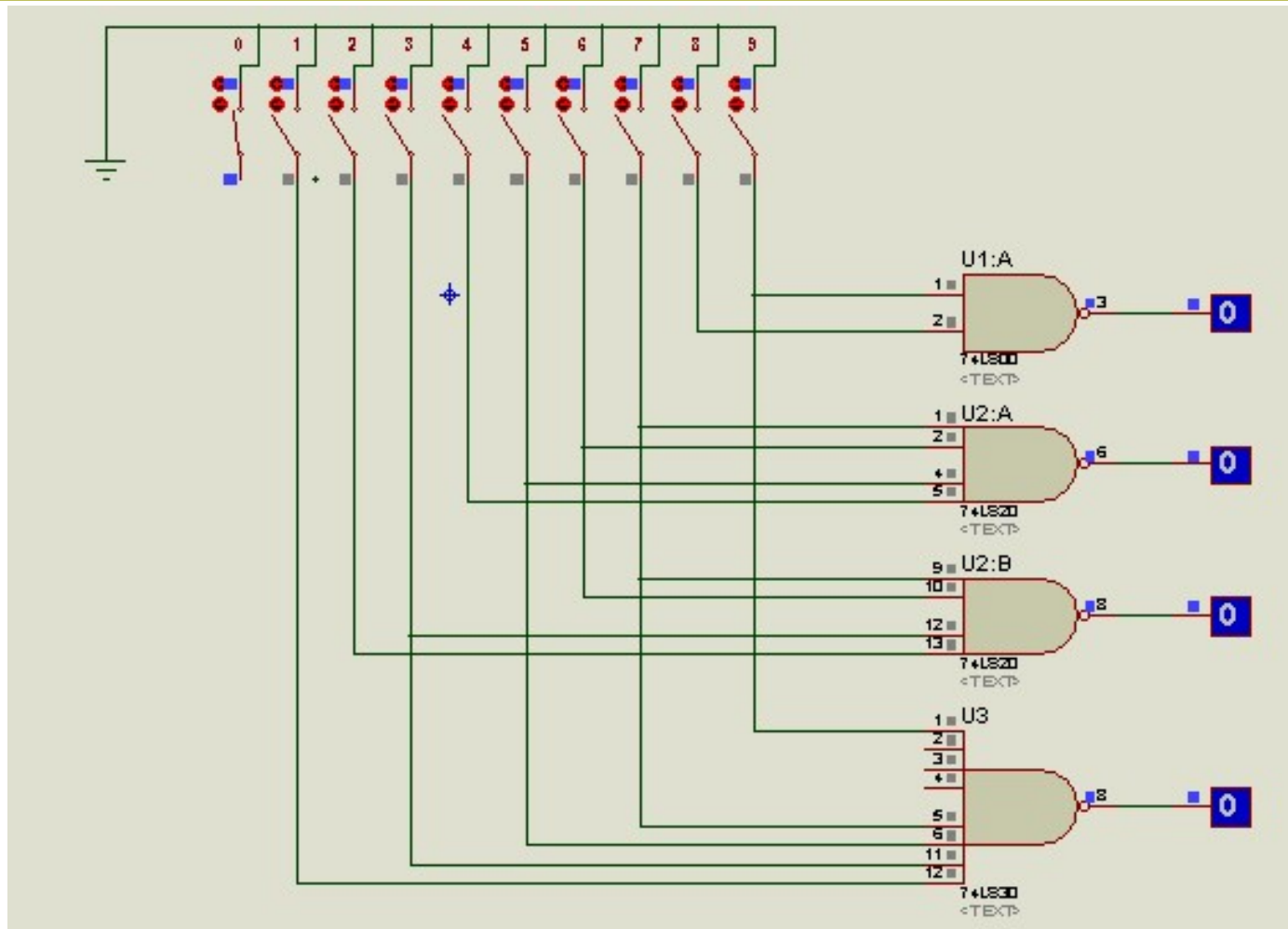
Tabela do Codificador

Tabela da verdade				
Relação da entrada decimal com a saída em binário				
Chave	A	B	C	D
Ch0	0	0	0	0
Ch1	0	0	0	1
Ch2	0	0	1	0
Ch3	0	0	1	1
Ch4	0	1	0	0
Ch5	0	1	0	1
Ch6	0	1	1	0
Ch7	0	1	1	1
Ch8	1	0	0	0
Ch9	1	0	0	1

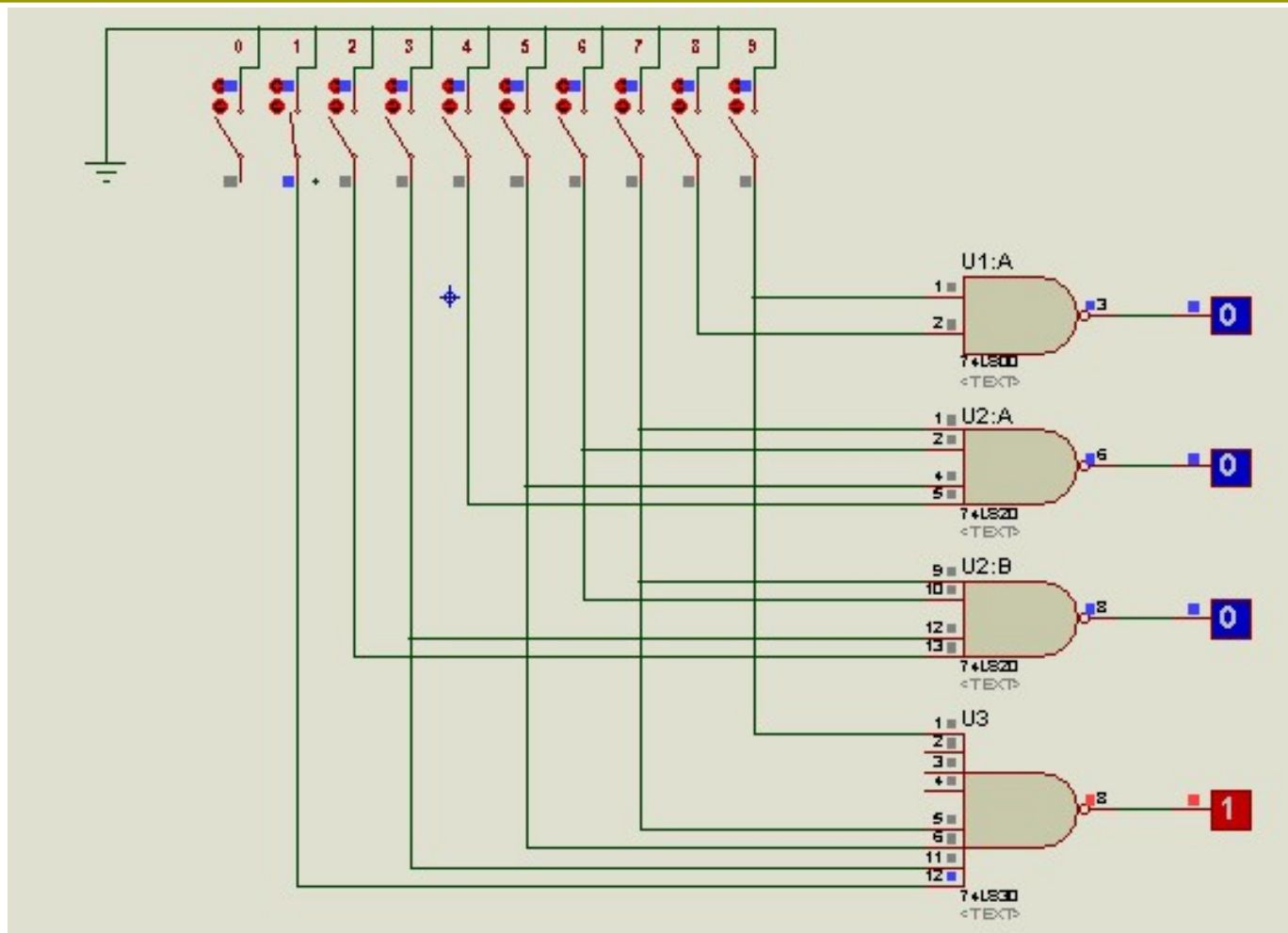
Circuito codificador



Funcionamento do codificador – Chave 0



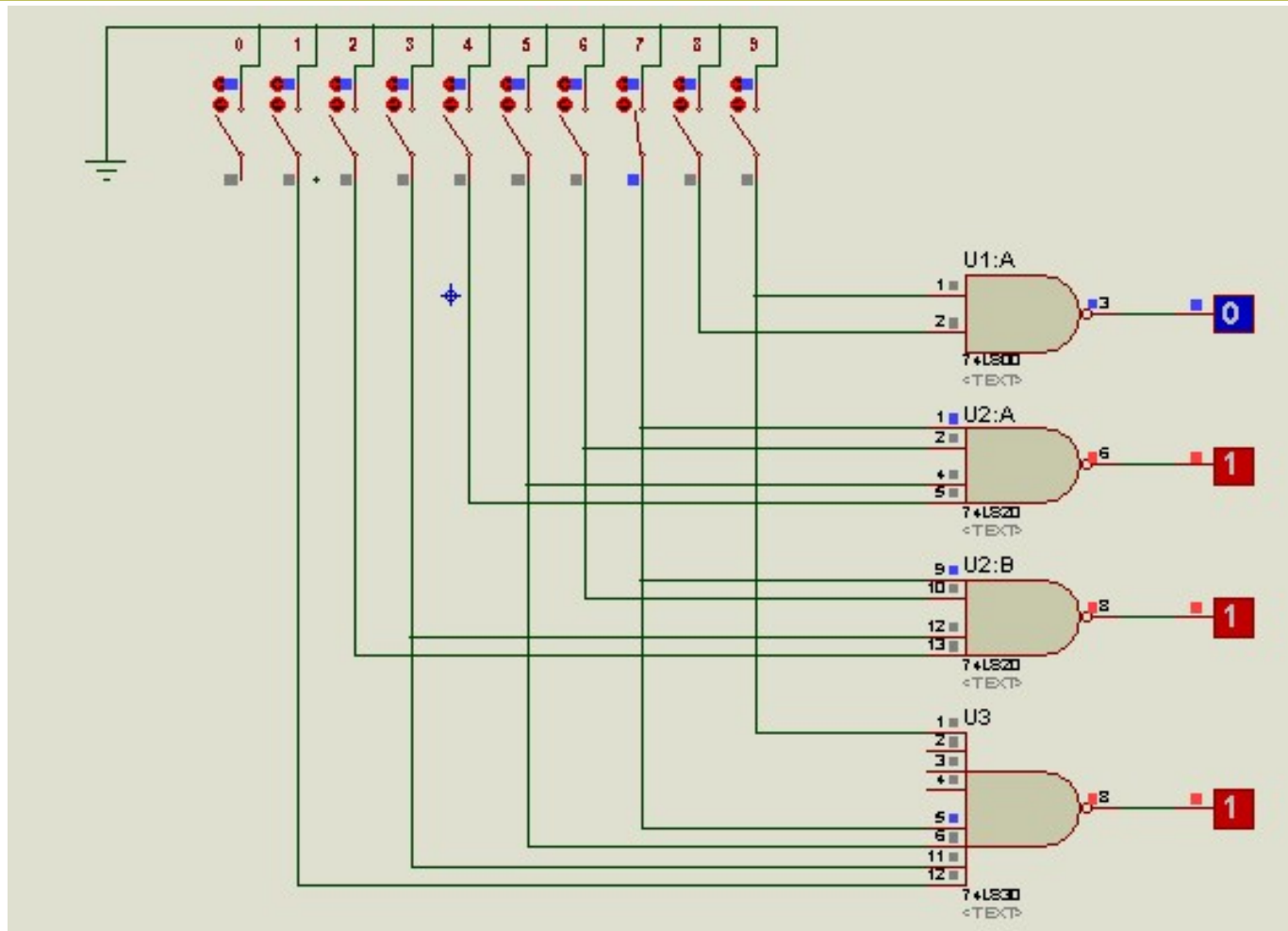
Funcionamento do codificador – Chave 1



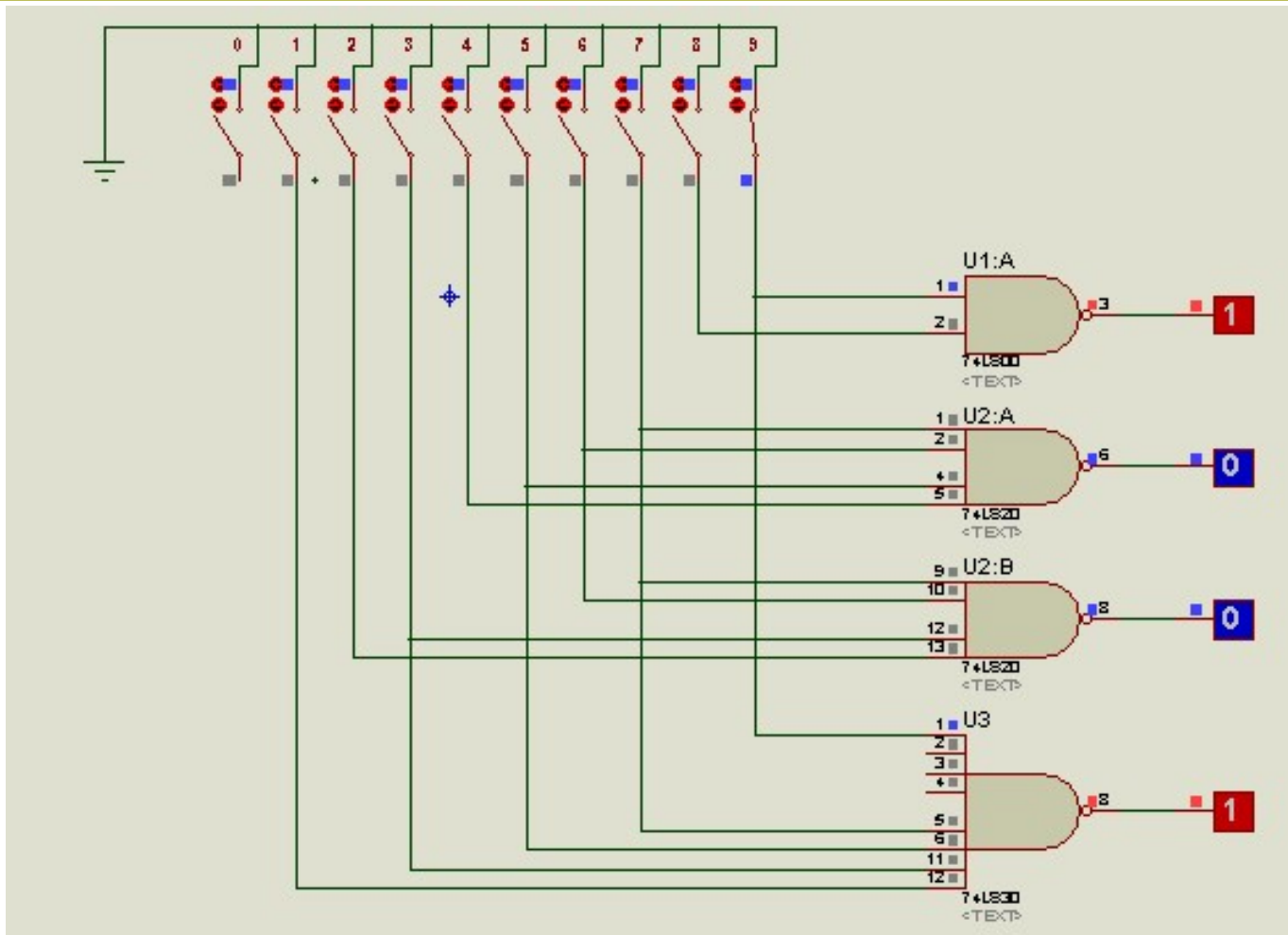
BMS

BMS

Funcionamento do codificador – Chave 7



Funcionamento do codificador – Chave 9



Circuito integrado TTL 74147

Codificador Decimal-BCD

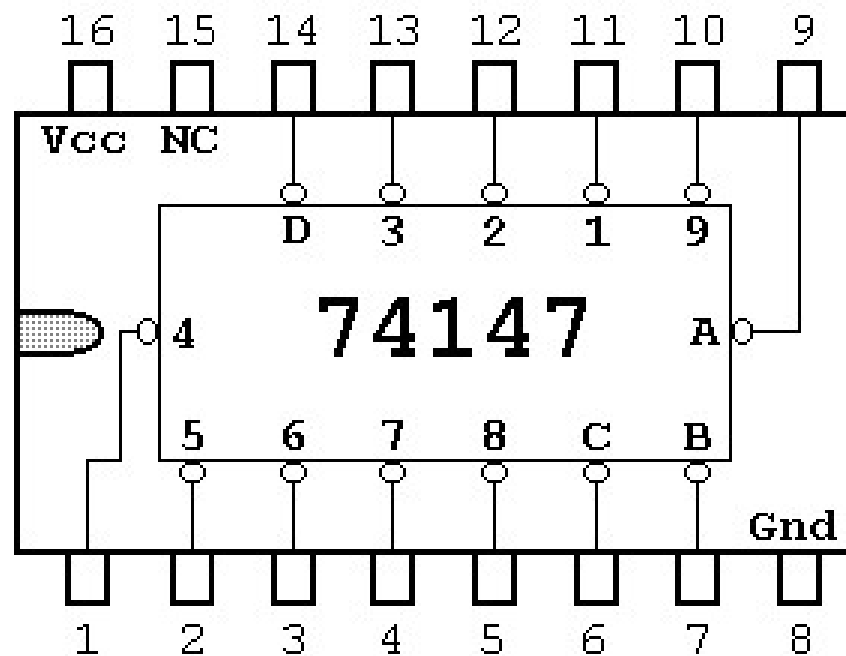
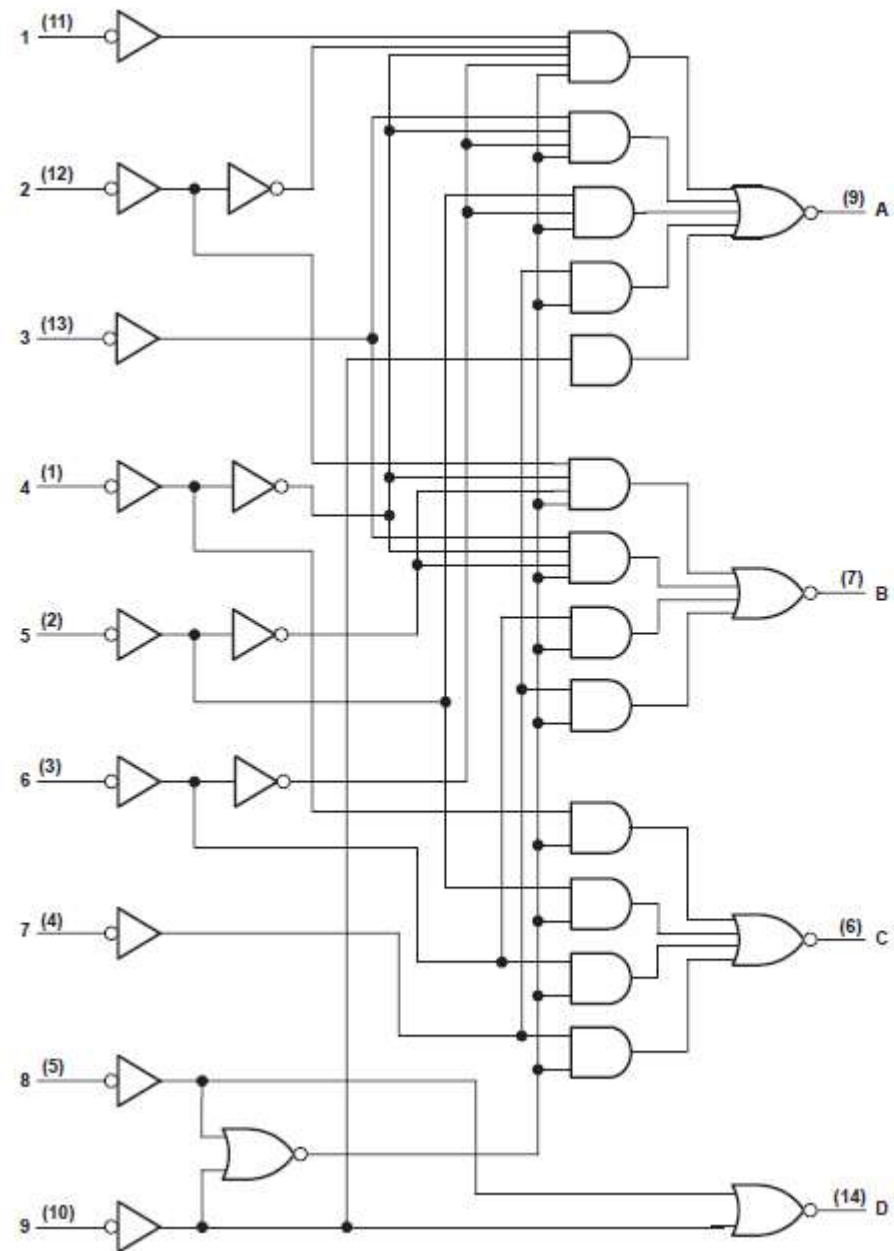


Tabela da verdade do CI 74xx147

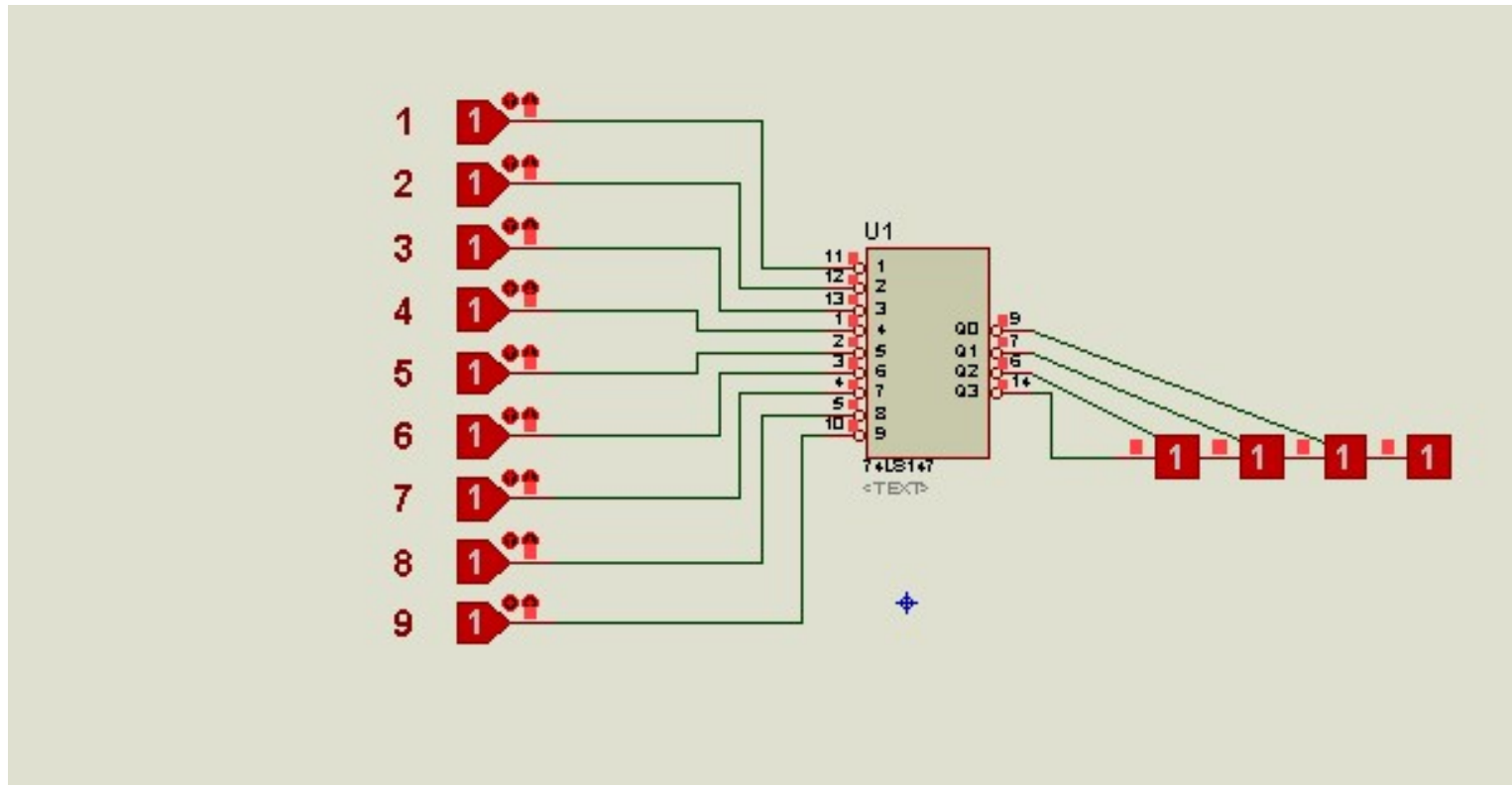
FUNCTION TABLE - '147, 'LS147

INPUTS									OUTPUTS			
1	2	3	4	5	6	7	8	9	D	C	B	A
H	H	H	H	H	H	H	H	H	H	H	H	H
X	X	X	X	X	X	X	X	L	L	H	H	L
X	X	X	X	X	X	X	L	H	L	H	H	H
X	X	X	X	X	X	L	H	H	H	L	L	L
X	X	X	X	X	L	H	H	H	H	L	L	H
X	X	X	X	L	H	H	H	H	H	L	H	L
X	X	X	L	H	H	H	H	H	H	L	H	H
X	X	L	H	H	H	H	H	H	H	H	L	L
X	L	H	H	H	H	H	H	H	H	H	L	H
L	H	H	H	H	H	H	H	H	H	H	H	L

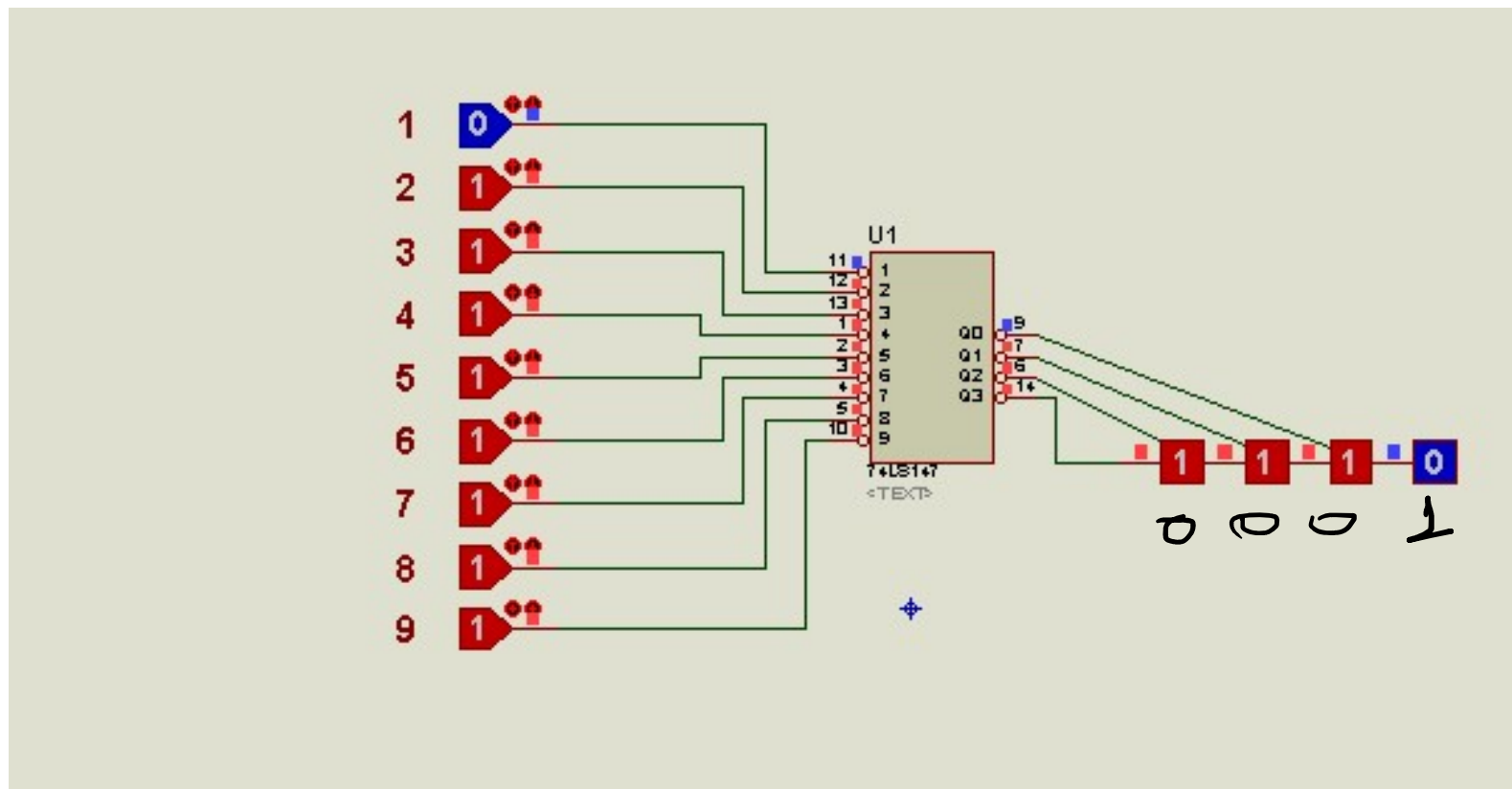
H = high logic level, L = low logic level, X = irrelevant

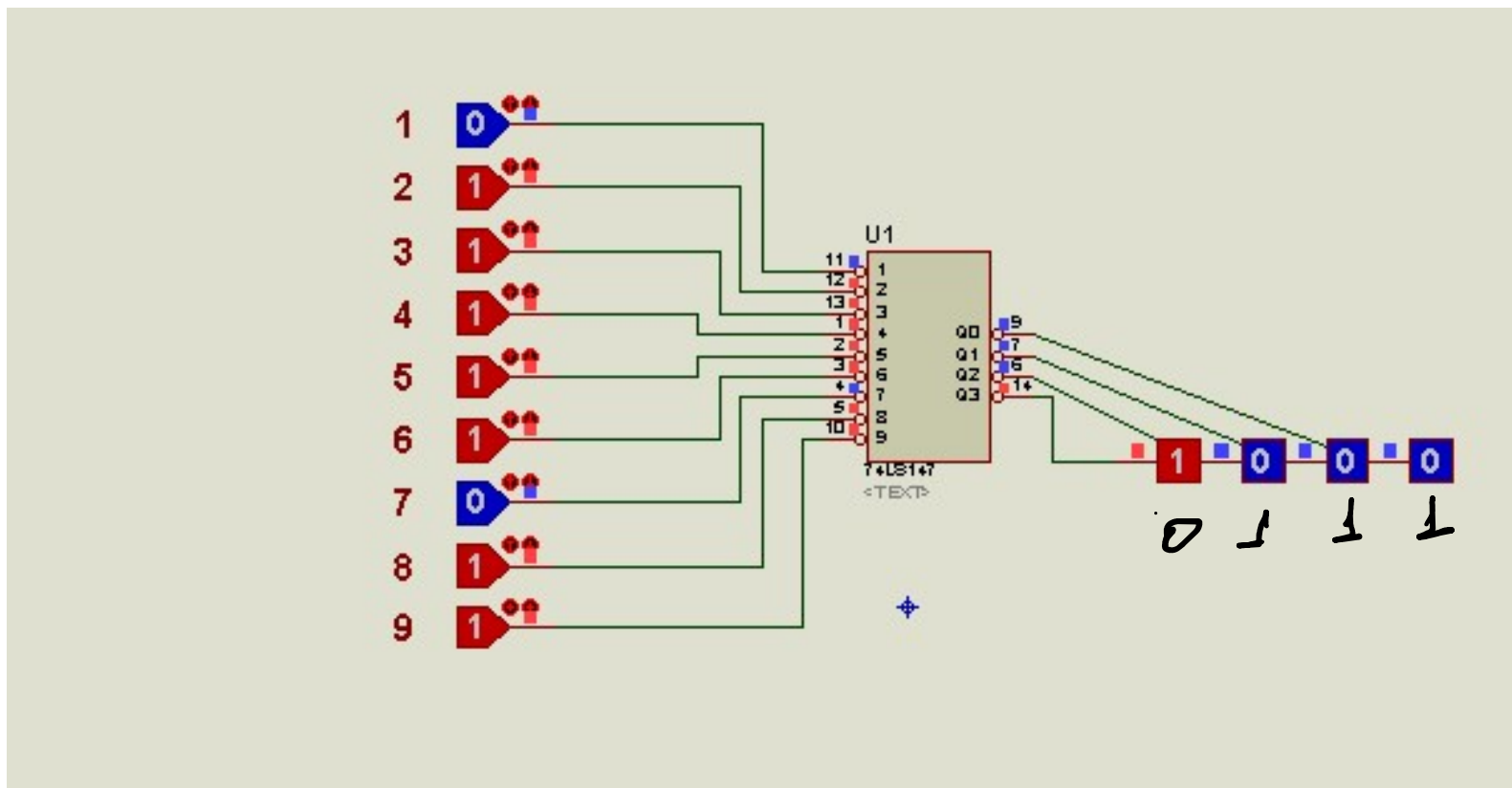


Teste do CI



Todas as entradas e todas as saídas são invertidas.

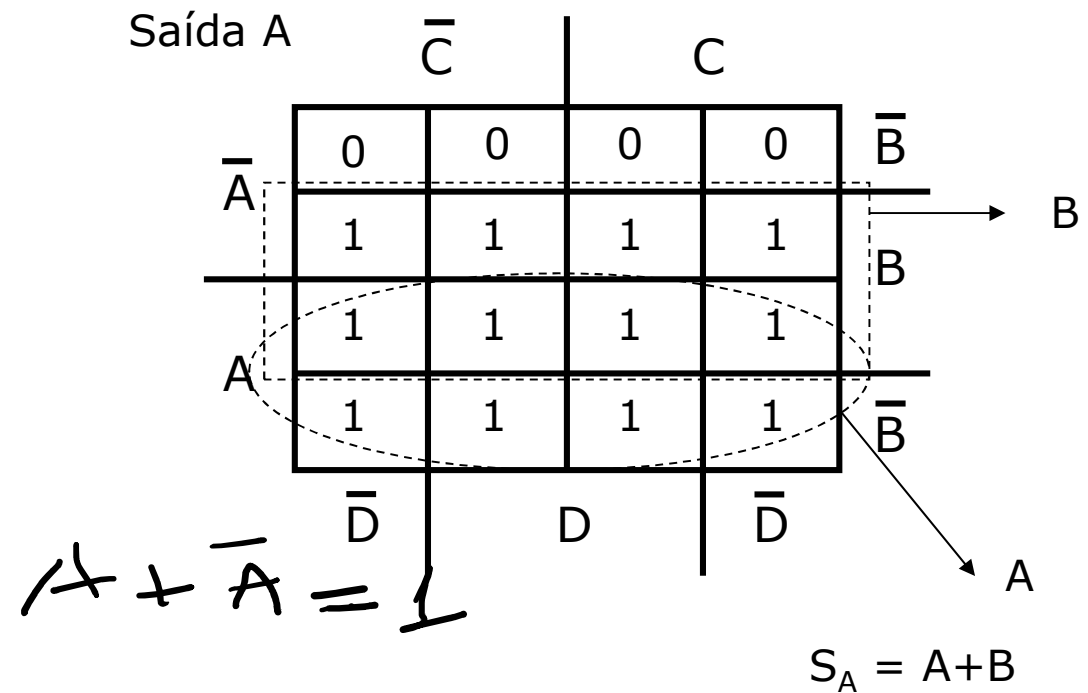
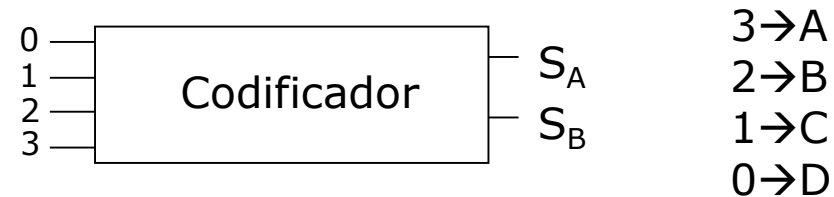




Quando duas ou mais entradas são acionadas, é mostrado o valor mais alto.

Ex. Elabore um codificador decimal/binário para, a partir de um teclado com chaves numeradas de 0 a 3, fornecer nas saídas o código binário correspondente. Considere a chave acionada como nível lógico 1, e que se tiver mais de uma chave acionada, mostrar o valor mais alto.

3210	$S_A S_B$
0000	0 0
0001	0 0
0010	0 1
0011	0 1
0100	1 0
0101	1 0
0110	1 0
0111	1 0
1000	1 1
1001	1 1
1010	1 1
1011	1 1
1100	1 1
1101	1 1
1110	1 1
1111	1 1



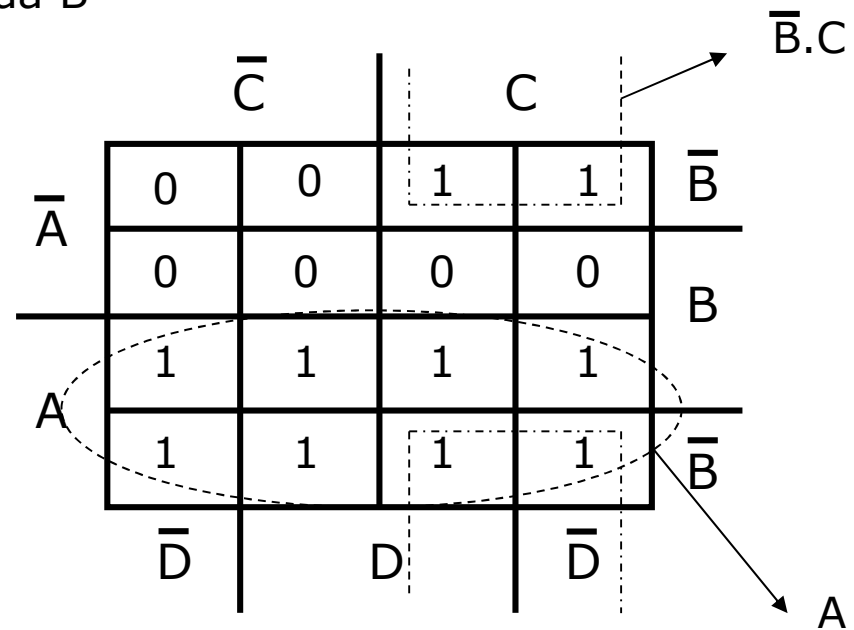
Exemplo

Elabore um codificador decimal/binário para, a partir de um teclado com chaves numeradas de 0 a 3, fornecer nas saídas o código binário correspondente. Considere a chave acionada como nível lógico 1, e que se tiver mais de uma chave acionada, mostrar o valor mais alto.



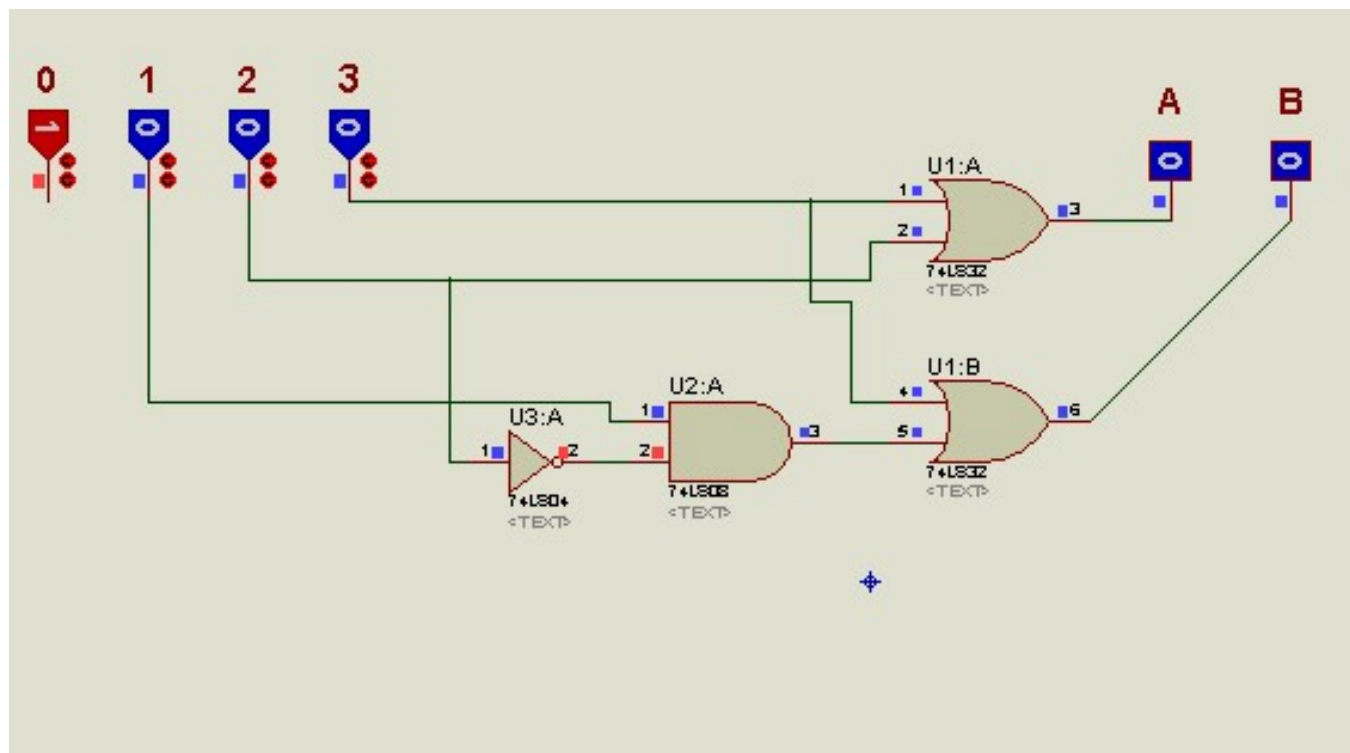
3210	AB
0000	00
0001	00
0010	01
0011	01
0100	10
0101	10
0110	10
0111	10
1000	11
1001	11
1010	11
1011	11
1100	11
1101	11
1110	11
1111	11

Saída B

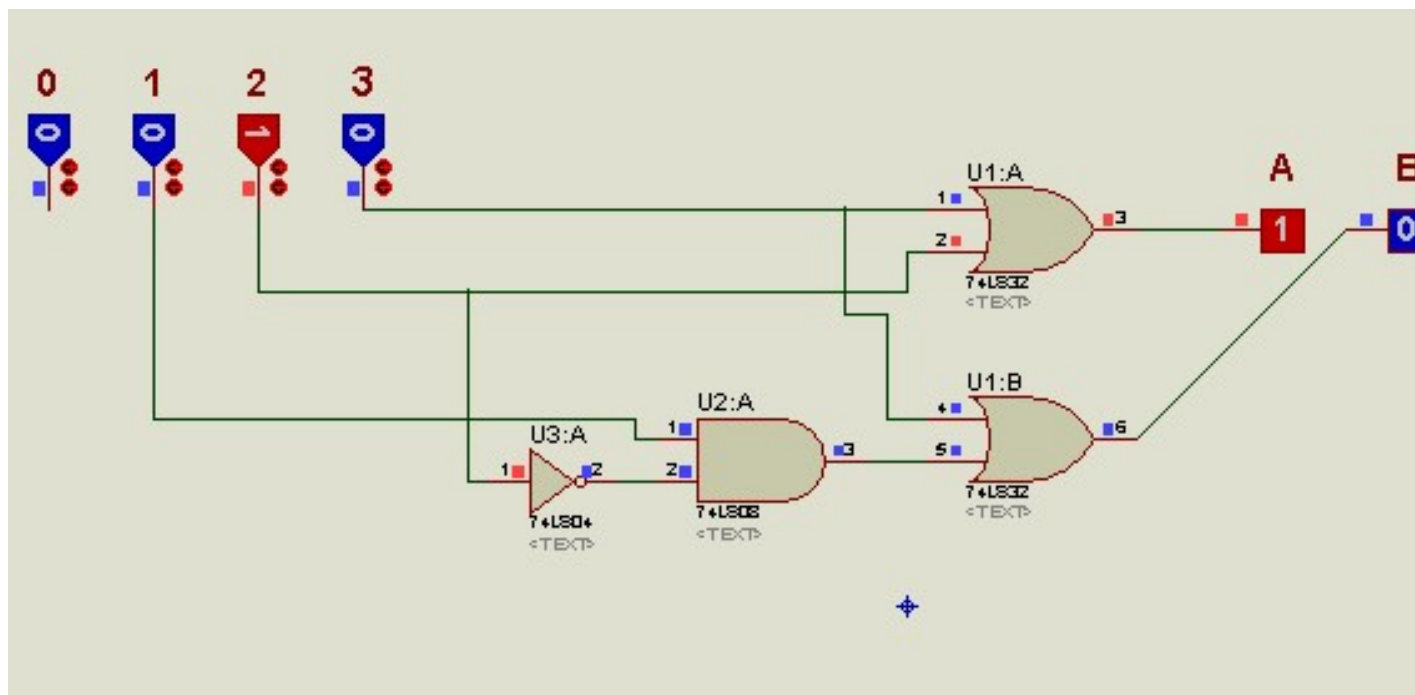


$$S_B = A + \bar{B}.C$$

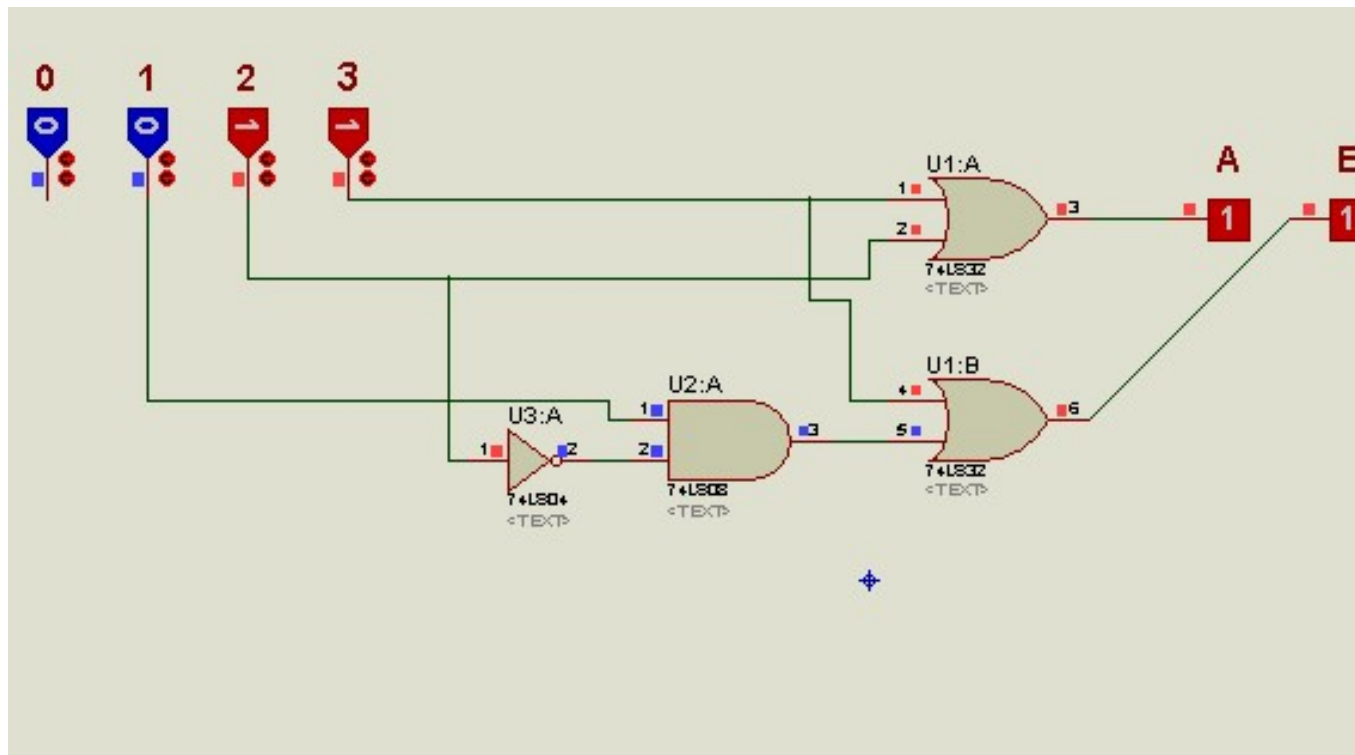
Circuito



Apertando a tecla "0", é mostrado "00" nos leds.

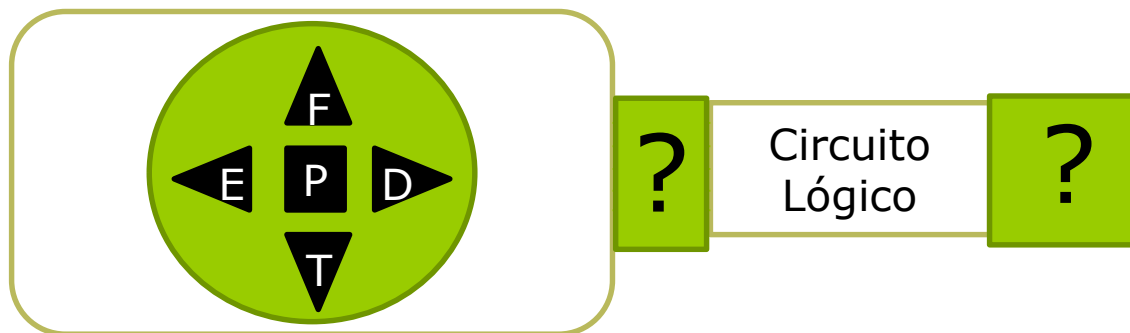


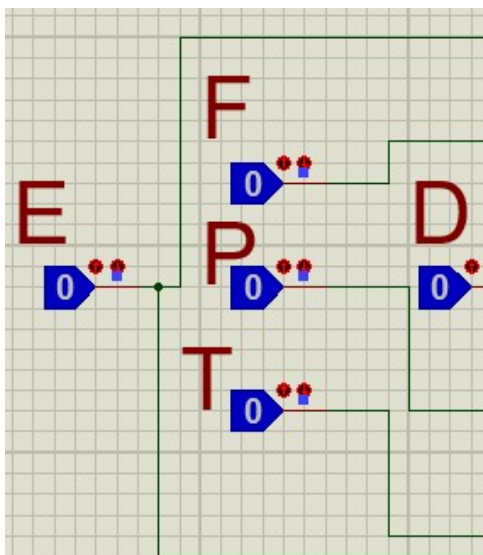
Apertando a tecla "1", é mostrado "01" nos leds, e apertando a tecla "2", é mostrado "10" nos leds.



Apertando a tecla "2" e a tecla "3", simultaneamente, é mostrado "11" nos leds, pois o projeto foi feito para mostrar o valor mais alto.

- Ex. Elabore um codificador para fornecer nas saídas o código binário correspondente aos botões conforme o desenho abaixo. Considere que se nenhum botão for acionado, o código de saída deverá ser zero, e se dois ou mais botões forem acionados simultaneamente, o código também deverá ser zero, e se a letra P for acionada também será zero.





Circuito lógico e as saídas



