UNIUBE – CAMPUS VIA CENTRO – Uberlândia/MG
Curso de Engenharia Elétrica e Engenharia de Computação
Disciplina: Sistemas Digitais

Aula 08
Codificadores, decodificadores e multiplexadores

Revisão 2, de 27/03/2025

Prof. João Paulo Seno
joao.seno@uniube.br

1

Decodificador

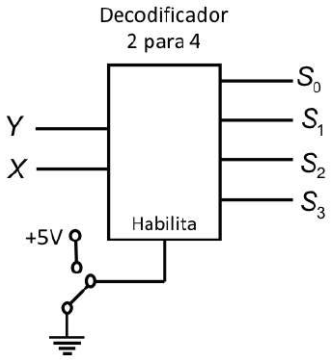
- Esse tipo de circuito digital determina uma saída específica para cada combinação única de bits ou código de entradas.
- De uma forma geral, um decodificador converte as informações binárias de *n* linhas de entradas para um máximo de 2^n linhas únicas de saída.
- Por exemplo, um decodificador com duas linhas de entrada (2 bits, X, Y) consegue, no máximo, endereçar $2^2 = 4$ linhas únicas de saída (S_3, S_2, S_1, S_0).
- Veja o próximo *slide*.

Uniube

Decodificador

Decodificador 2 para 4

Habilita	Y	X	S_0	S_1	S_2	S_3
0	X	X	0	0	0	0
1	0	0	1	0	0	0
1	0	1	0	1	0	0
1	1	0	0	0	1	0
1	1	1	0	0	0	1



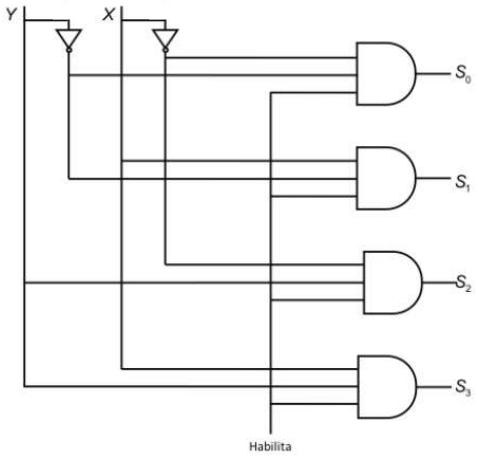
Uniube

Entrada “habilita” (*Enable*)

- Note, no circuito da figura anterior, a presença da entrada adicional “habilita” (ou *Enable*).
- Esta entrada, quando em nível alto, determina que a entrada do decodificador esteja ativada ou seja, habilitada para determinar as saídas especificadas pelo código dos bits de entrada.
- Porém, quando a entrada “habilita” está em nível baixo, a entrada do decodificador estará desativada e não importa (*don’t care*) quais forem os bits de entrada (Y, X), pois as saídas estarão sempre em nível baixo.

Uniube

Implementação da entrada “habilita” (*enable*)

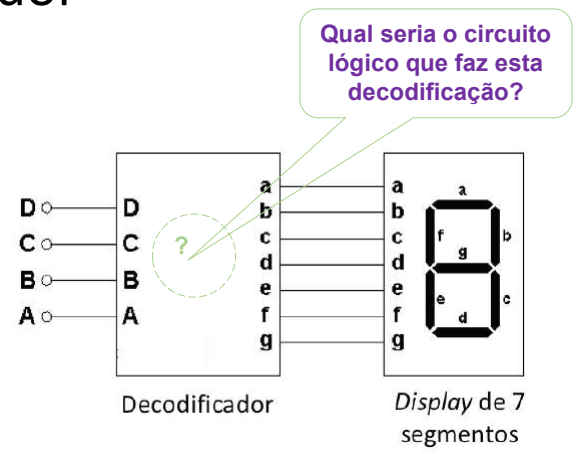


Uniube

Exemplo de decodificador

BCD para 7 segmentos

Decimal	Binário	BCD
0	0000	0000
1	0001	0001
2	0010	0010
3	0011	0011
4	0100	0100
5	0101	0101
6	0110	0110
7	0111	0111
8	1000	1000
9	1001	1001



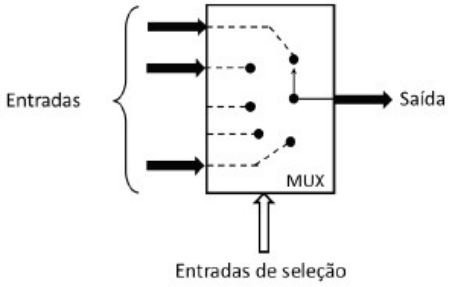
- Você saberia como implementar o decodificador BCD para 7 segmentos, apresentado acima?

Uniube

Multiplexador

Definição

- Multiplexador (ou *Mux*) digital é um circuito lógico que recebe diversos dados digitais de entrada e seleciona um, em determinado instante para transferi-lo à saída.
- O envio do dado é controlado pelas entradas de seleção ou controle.



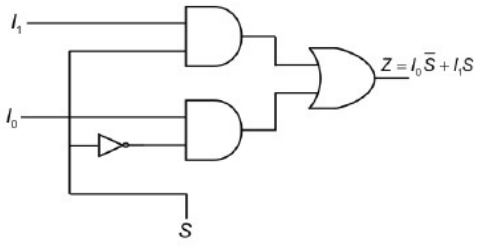
Uniube

Multiplexador

- O circuito multiplexador básico possui duas entradas (***I*₀ e *I*₁**) de dados.
- Portanto é necessária apenas uma linha de seleção (***S***).
- Observe o circuito lógico da figura abaixo. O nível lógico aplicado à chave seletora determina a porta AND a ser habilitada.
- A resposta dessa porta lógica é enviada a uma porta OR para então o canal de dado selecionado ser expresso na saída (***Z***). Veja a função lógica que foi implementada!

Tabela verdade

S	Saída
0	I_0
1	I_1



Uniube

São necessárias ***n*** linhas de seleção para 2^n dados de entrada.

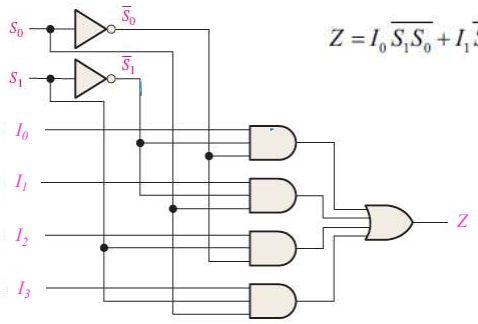
Uniube

Multiplexador de 4 entradas

- Para esse circuito, são necessárias duas linhas de seleção pois, com dois bits, qualquer uma das quatro linhas de entrada de dados pode ser selecionada. Observe a função lógica ***Z***, que está implementada.

Tabela verdade

S_1	S_0	Entrada Selecionada
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

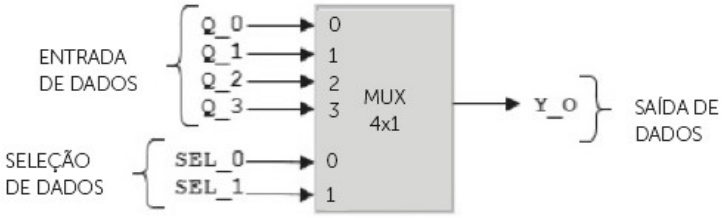


$$Z = I_0\overline{S_1}\overline{S_0} + I_1\overline{S_1}S_0 + I_2S_1\overline{S_0} + I_3S_1S_0$$

Uniube

Multiplexador de 4 entradas

- Uma outra representação, encontradas em alguns livros:



Uniube

A entrada *Enable* (***E***)

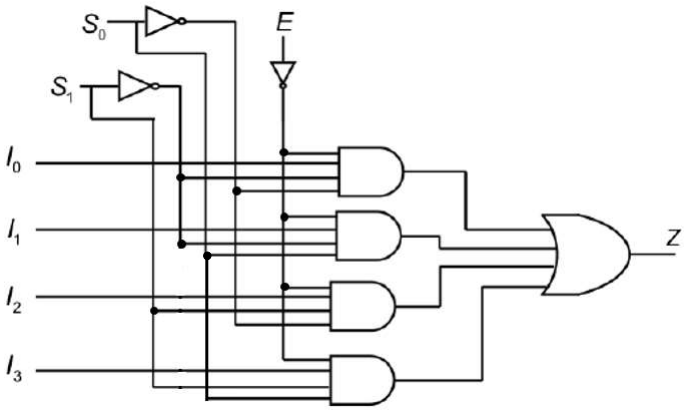
- Além das entradas de dados, de seleção e saída, os CIs que contêm os mutiplexadores também possuem uma entrada denominada *habilitação* ou *Enable*, em inglês (***E***).
- Esta entrada habilita a saída, ou seja, quando é aplicado o nível lógico correto, é permitido que o dado da entrada selecionada passe para a saída. Caso contrário, nenhum dado é visto na saída do circuito multiplexador, independentemente do código na entrada de seleção, isto é, a saída está desabilitada.
- Já vimos esta entrada nos circuitos combinacionais do tipo codificador e decodificador na seção anterior.

Uniube

Circuito *Mux* de 4 entradas com *Enable* (*E*)

Implementação

E	S_1	S_0	Entrada Selecionada
0	0	0	I_0
1	0	0	0
0	0	1	I_1
1	0	1	0
0	1	0	I_2
1	1	0	0
0	1	1	I_3
1	1	1	0



Observe a função Z que foi implementada!

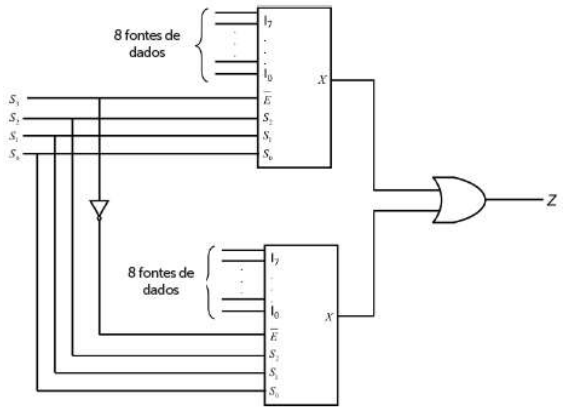
$$Z = \overline{E} I_0 \overline{S_0}\,\overline{S_1} + \overline{E} I_1 \overline{S_1} S_0 + \overline{E} I_2 S_1 \overline{S_0} + \overline{E} I_3 S_1 S_0$$

Uniube

Situação prática

Mas entradas do que um CI permite!

- Suponha que você tem 16 fontes de dados diferentes como entrada e apenas CIs multiplexadores com oito entradas. Como você poderia enviar esses dados em apenas um canal de comunicação? Veja o circuito lógico abaixo. Entendeu a solução encontrada?

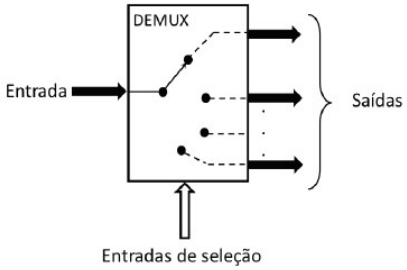


Uniube

Demultiplexador

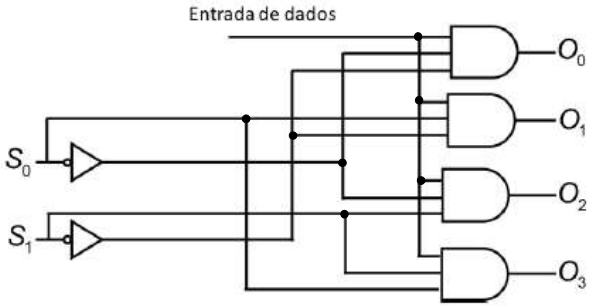
Definição

- Os circuitos combinacionais demultiplexador (*Demux*) realizam a operação inversa ao mux. Os circuitos classificados como *demux* recebem uma única entrada e a distribuem para várias saídas.
- Por essa razão, o demultiplexador também é conhecido como distribuidor de dados. Veja o diagrama funcional do Demux, abaixo.

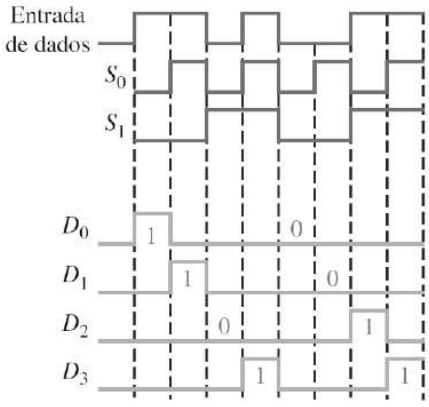


Uniube

Diagrama lógico de um demultiplexador de uma linha para quatro linhas



Exemplo de aplicação do *Demux* de 4 saídas



Uniube

Fim