Universidade Federal de Campina Grande
Centro de Engenharia Elétrica e Informática
Unidade Acadêmica de Sistemas e Computação
Curso de Bacharelado em Ciência da Computação

Organização e Arquitetura de Computadores

Circuitos Lógicos Combinacionais (Parte II)

Profa. Joseana Macêdo Fechine Régis de Araújo
joseana@computacao.ufcg.edu.br

Carga Horária: 60 horas

Tópicos

Circuitos Lógicos Combinacionais

- Decodificadores
- Codificadores
- Multiplexadores
- Demultiplexadores
- Gerador/Verificador de Paridade

Projeto de Circuitos Combinacionais

Decodificação

- Conversão de um código de entrada de n-bits em um código de saída de m bits com n ≤ m ≤ 2^n tal que cada palavra-código válida produz um único código de saída.

Decodificador

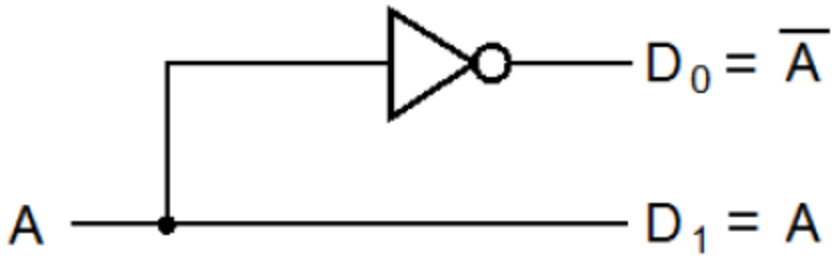
- Circuito que implementa a decodificação.

Projeto de Circuitos Combinacionais

Exemplo: Decodificador de 1-2 linhas

A	D_0	D_1
0	1	0
1	0	1

(a)



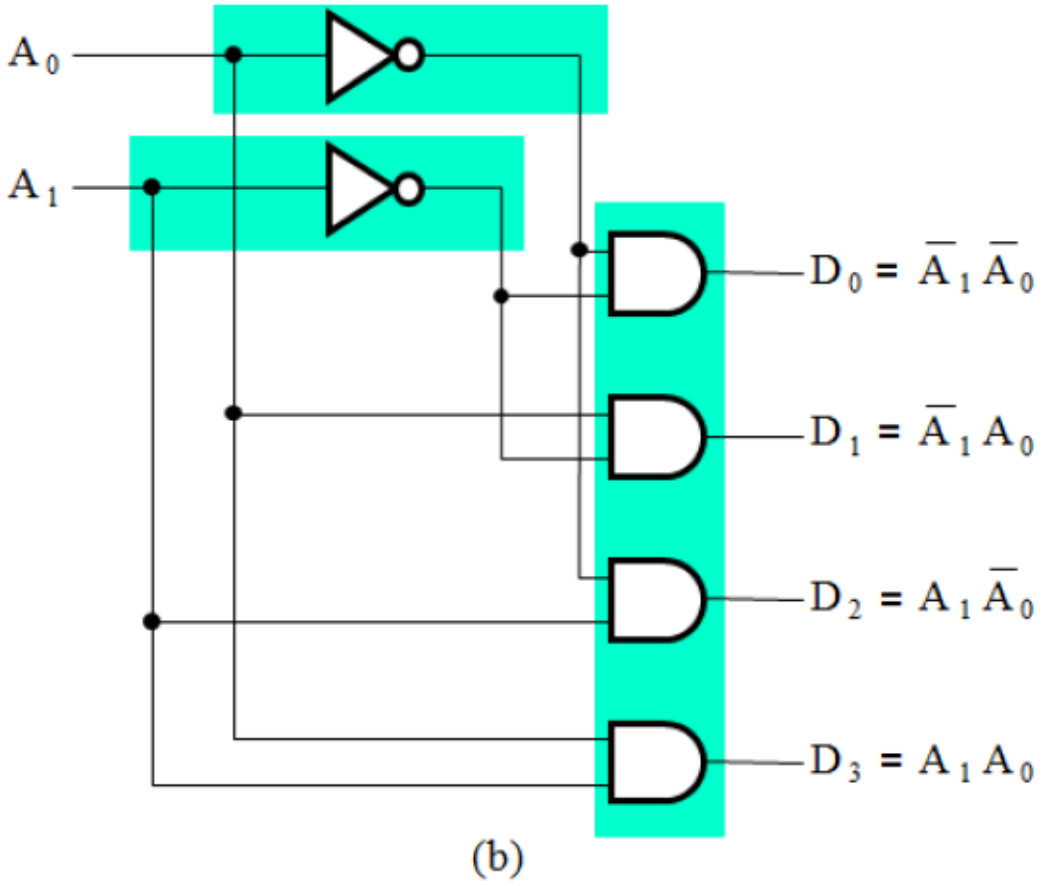
(b)

Projeto de Circuitos Combinacionais

Exemplo: Decodificador de 2-4 linhas

A_1	A_0	D_0	D_1	D_2	D_3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

(a)



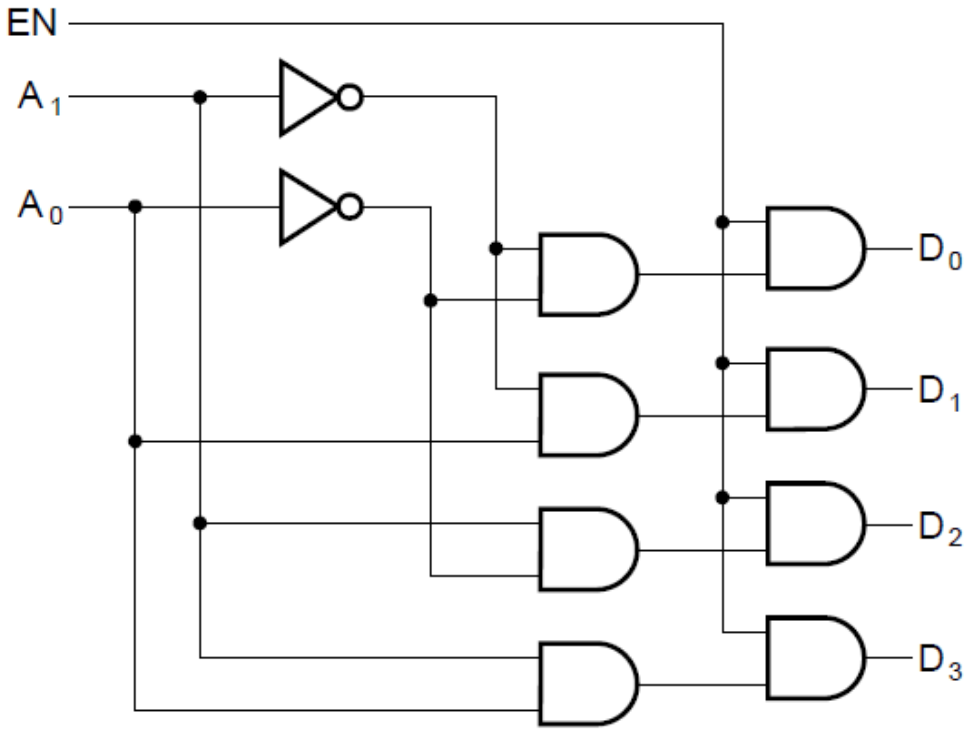
(b)

Projeto de Circuitos Combinacionais

Exemplo: Decodificador com *Enable*

EN	A_1	A_0	D_0	D_1	D_2	D_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

(a)



(b)

Projeto de Circuitos Combinacionais

Decodificadores

(outras formas)

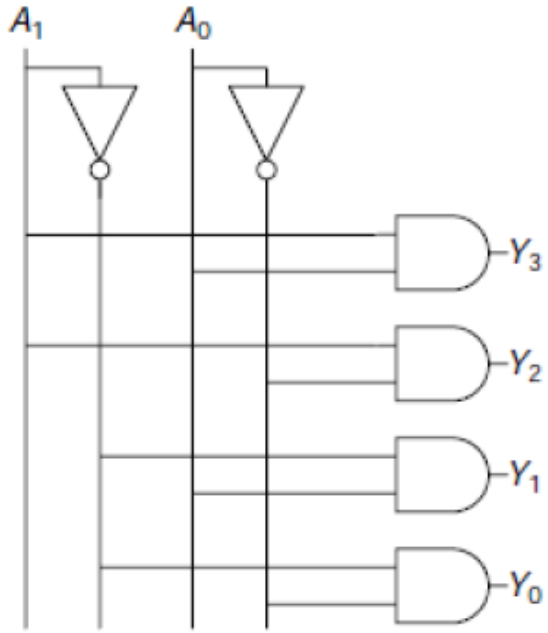
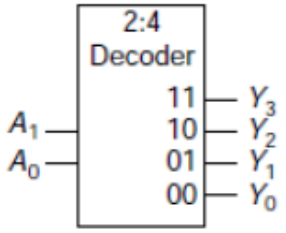


Figure 2.64 2:4 decoder implementation



A_1	A_0	Y_3	Y_2	Y_1	Y_0
0	0	0	0	0	1
0	1	0	0	1	0
1	0	0	1	0	0
1	1	1	0	0	0

Figure 2.63 2:4 decoder

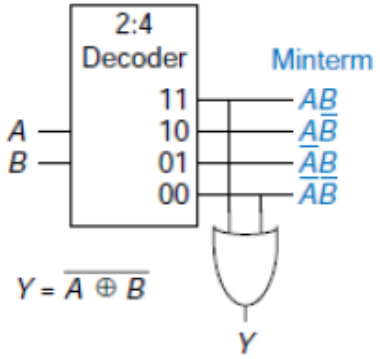


Figure 2.65 Logic function using decoder

Projeto de Circuitos Combinacionais

Codificação

- O oposto da decodificação - conversão de um código de entrada de m-bits em um código de saída de n bits com com $n \leq m \leq 2^n$, tal que cada palavra-código válida produz um único código de saída.

Codificador

- Circuito que implementa a codificação.

Exemplo: Codificador decimal para BCD

- Entradas: 10 bits correspondente aos dígitos decimais de 0 a 9, (D0, ... D9)

Projeto de Circuitos Combinacionais

Decodificadores e Codificadores – outras aplicações

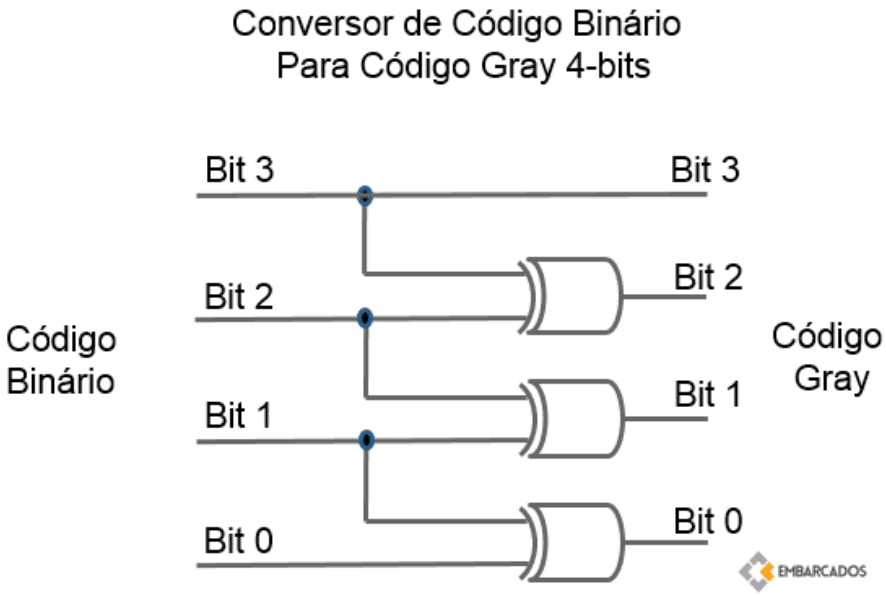
- Grande parte dos sistemas digitais trabalha com níveis lógicos representando informações que, portanto, devem ser **codificadas**.

- Outros códigos: Código Gray, Código 2 em 5, Código em Anel, Código para acionamento de *display*, etc..

Projeto de Circuitos Combinacionais

Decodificadores e Codificadores – outras aplicações

DECIMAL	BINARIO	GRAY
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1110
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000



Projeto de Circuitos Combinacionais

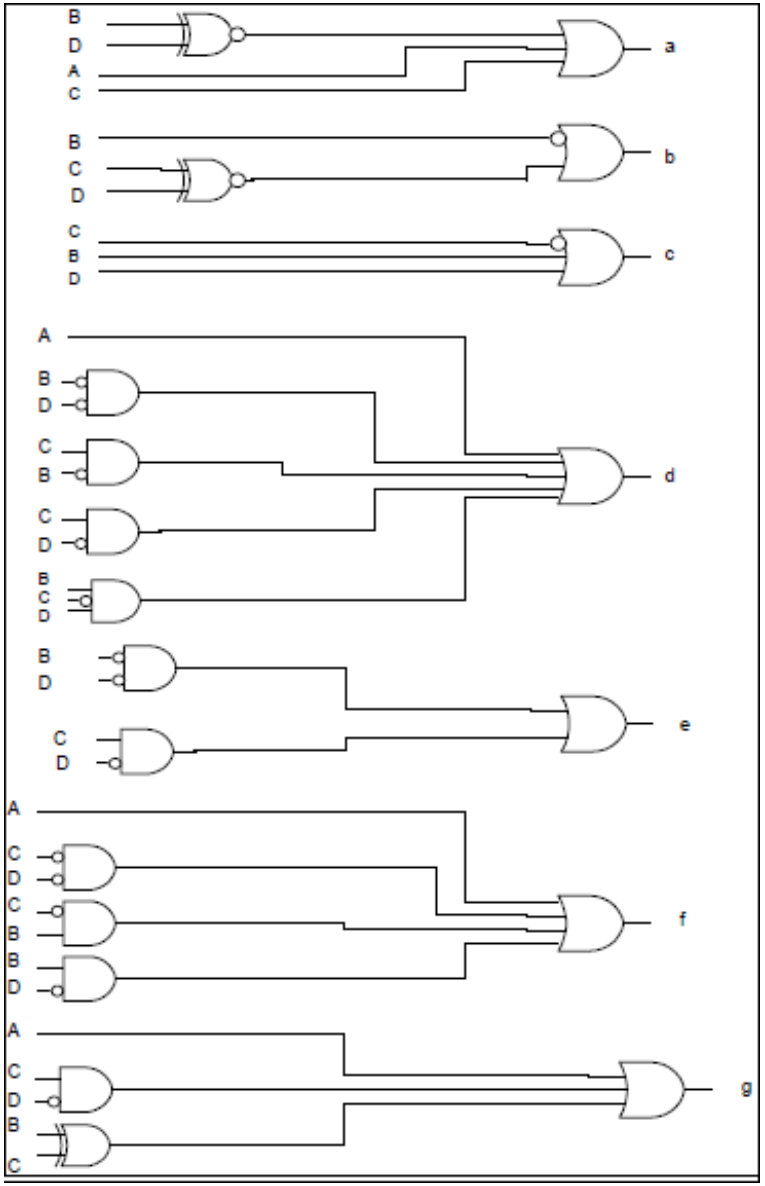
Exemplo: Decodificador BCD - Display de 7 segmentos

Projeto de Circuitos Combinacionais

Exemplo: Decodificador BCD - Display de 7 segmentos

Entrada (BCD/Binário)				*Saídas 7 segmentos*							*Display*
D	C	B	A	*a*	*b*	*c*	*d*	*e*	*f*	*g*	
0	0	0	0	1	1	1	1	1	1	0	0
0	0	0	1	0	1	1	0	0	0	0	1
0	0	1	0	1	1	0	1	1	0	1	2
0	0	1	1	1	1	1	1	0	0	1	3
0	1	0	0	0	1	1	0	0	1	1	4
0	1	0	1	1	0	1	1	0	1	1	5
0	1	1	0	0	0	1	1	1	1	1	6
0	1	1	1	1	1	1	0	0	0	0	7
1	0	0	0	1	1	1	1	1	1	1	8
1	0	0	1	1	1	1	0	0	1	1	9

www.arduinoecia.com.br



Projeto de Circuitos Combinacionais

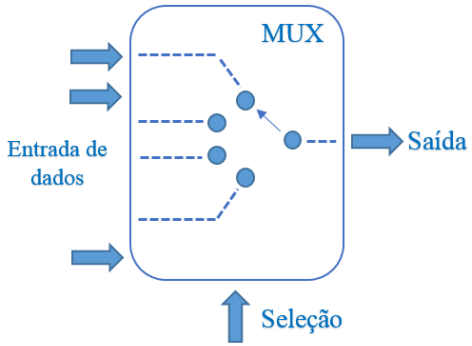
Multiplexadores: Circuitos lógicos que realizam seleção.

- Seleção de dados ou informação: uma operação crítica em sistemas digitais e computadores.
- Características de um circuito seletor:
  - Um conjunto de informações de entrada para as quais será feita a seleção;
  - Uma saída; e
  - Um conjunto de sinais de controle responsáveis por selecionar a informação de saída.

Projeto de Circuitos Combinacionais

- **Multiplexadores (MUX):** circuitos combinacionais que têm a finalidade de **selecionar**, a partir das **variáveis de seleção**, **uma** de suas **entradas**, conectando-a eletronicamente a sua **única saída**.

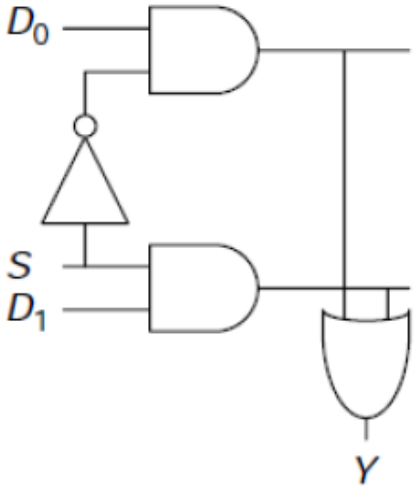
- Circuito com 2^n entradas de dados, uma saída de dados e ***n*** entradas de controle para efetuar a seleção de uma das entradas de dados.

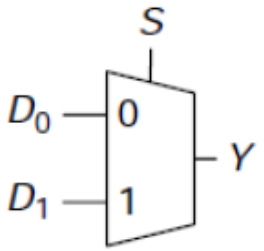


Projeto de Circuitos Combinacionais

- **Exemplo: MUX(2:1)**

Variável de seleção (S)	Saída
0	D_0
1	D_1



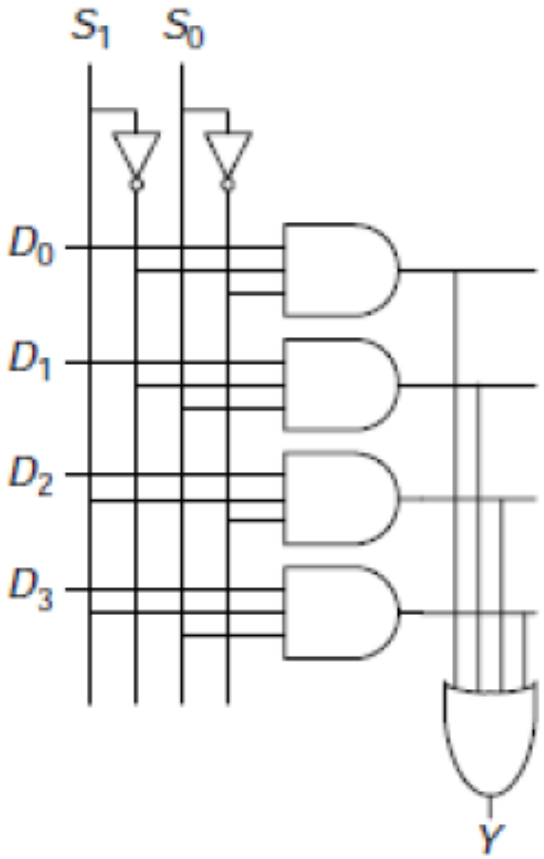


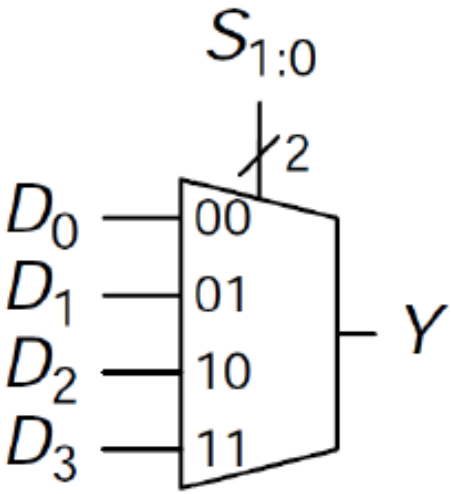
S	D_1	D_0	Y
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

Figure 2.54 2:1 multiplexer symbol and truth table

Projeto de Circuitos Combinacionais

- **Exemplo: MUX(4:1)**



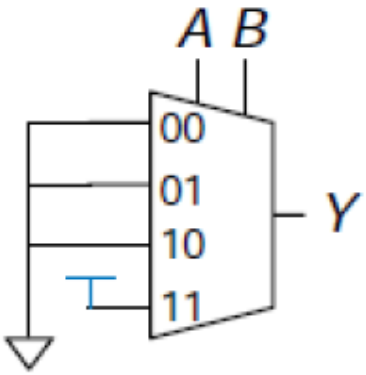


Projeto de Circuitos Combinacionais

Exemplo 1

A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

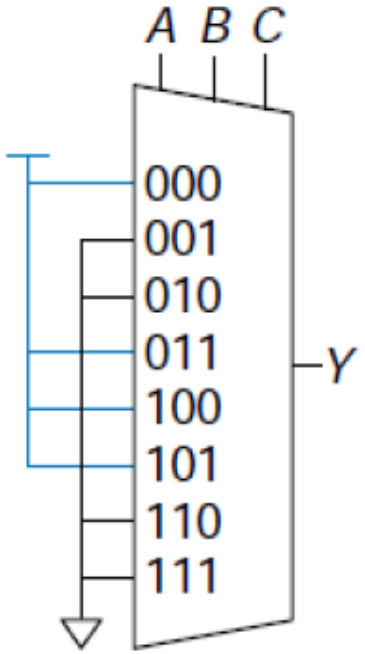
$Y = AB$



Exemplo 2

A	B	C	Y
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

$Y = A\overline{B} + \overline{B}\overline{C} + \overline{A}BC$



(b)

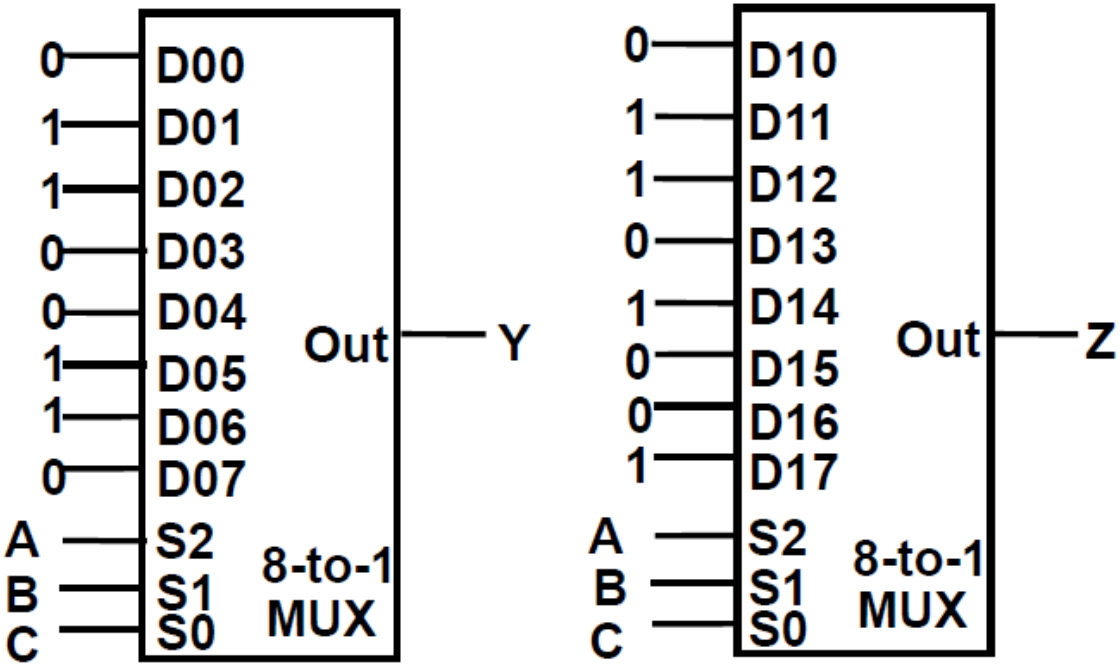
Projeto de Circuitos Combinacionais

- **Exemplo: Código Gray -> Código Binário usando multiplexadores**

Gray A B C	Binário X Y Z
0 0 0	0 0 0
1 0 0	0 0 1
1 1 0	0 1 0
0 1 0	0 1 1
0 1 1	1 0 0
1 1 1	1 0 1
1 0 1	1 1 0
0 0 1	1 1 1

Projeto de Circuitos Combinacionais

- **Código Gray -> Código Binário com MUX(8:1)**



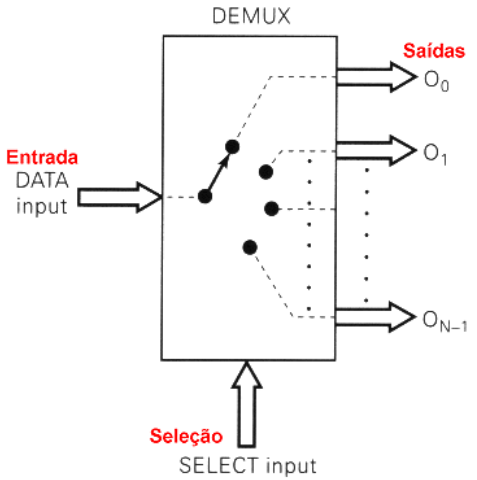
Obs.: O multiplexador com um número de entradas fixo é idêntico a uma **memória ROM** com 3 bits de endereço e 2 bits de dados.

Projeto de Circuitos Combinacionais

É possível obter o circuito Código Gray -> Código Binário utilizando MUX 2:1?

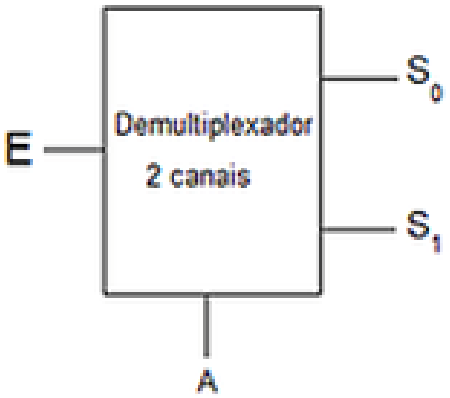
Projeto de Circuitos Combinacionais

- **Demultiplexadores (DEMUX):** têm a finalidade de **selecionar**, a partir das **variáveis de seleção**, qual de suas **saídas** deve receber a informação presente em sua **única entrada**

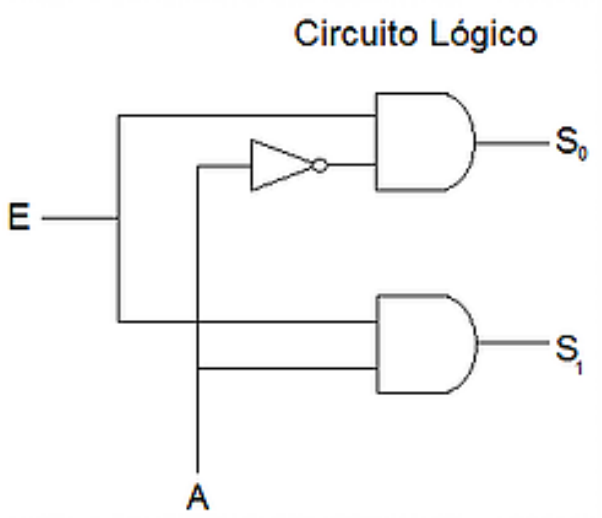


Projeto de Circuitos Combinacionais

- **Exemplo: (DEMUX 1:2)**



Variável de seleção	Canais de Saída S_0	S_1
0	E	0
1	0	E



Projeto de Circuitos Combinacionais

Circuito Gerador/Verificador de Paridade

- Utilizado para detectar erro em transmissão digital.
- Este processo pode ser vulnerável se houver mais do que um erro, permitindo assim que este passe até o destino sem ser identificado.
- Usado em muitas aplicações de hardware (em que uma operação pode ser repetida em caso de dificuldade, ou quando é útil a simples detecção de erros).

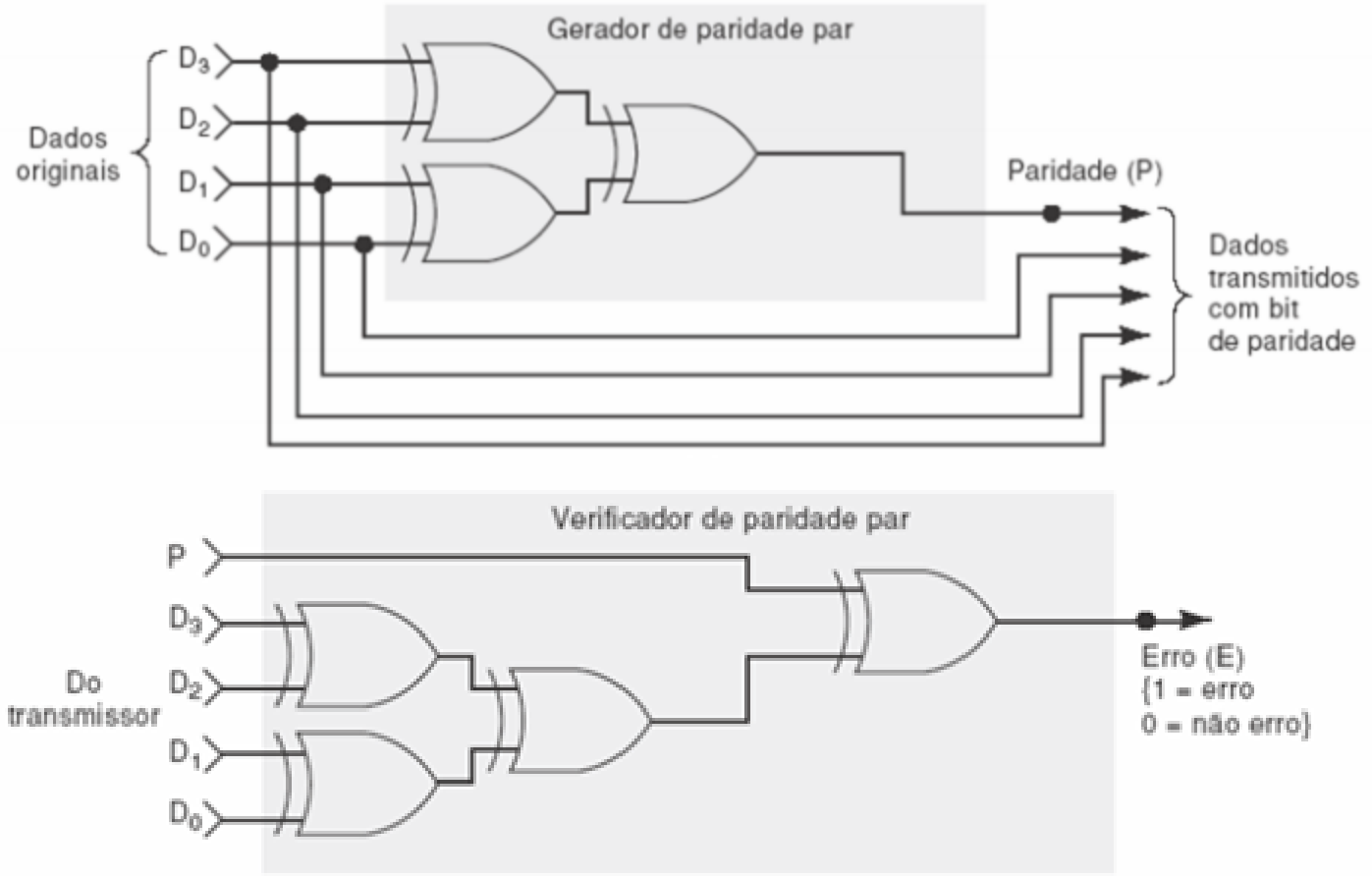
Projeto de Circuitos Combinacionais

Bit de Paridade

- Bit extra anexado ao conjunto de bits para informar a sua paridade.
- O bit de paridade pode ser 0 ou 1, dependendo do número de 1´s contido no conjunto de bits do código (par ou ímpar).
- **Paridade Par**: o bit anexado serve para tornar o número total de bits “1” **par** (Ex.: 01001 -> **0**01001).
- **Paridade Ímpar**: o bit anexado serve para tornar o número total de bits “1” **ímpar** (Ex.: 01001-> **1**01001).

Projeto de Circuitos Combinacionais

Exemplo: Geração e Verificação de Paridade Par (4 bits)



Projeto de Circuitos Combinacionais

Outras estratégias para detecção de erros

- **Checksum** - Consiste na transmissão de todas as palavras juntamente com o resultado da sua soma binária.
- **CRC** (***Cyclic Redundancy Check***).
- **Códigos de Hamming** - Detecção e Correção de Erros.