# Aula 0:

# Introdução e apresentação do curso

# Lógica Digital

## Objetivos

Compreender os fundamentos da eletrônica digital, desde a constituição de portas lógicas até circuitos sequenciais simples.

## Conteúdo programático

- Fundamentos de eletricidade
- Transistores
- Portas lógicas
- Circuitos integrados
- Expressões booleanas
- Tabelas-verdade
- Simuladores
- Álgebra de Boole
- Formas canônicas
- Mapas de Karnaugh
- Representação binária
- Displays e decoders
- Multiplexadores
- Somadores/subtratores
- Latches SR
- Sinais de clock
- Flip Flops SR, D, T, JK
- Registradores simples
- Registradores de deslocamento
- Contadores decrescentes

# Lógica Digital

## Carga horária

3h40

## Temas que ficaram de fora

- Linguagem de descrição de hardware e FGPAs: serão tratados em um curso sobre FPGAs

- Abordagem prática para todas as aulas: problemas de orçamento, câmera e tremor essencial.

- Máquinas de estado finito: serão tratadas em um curso sobre sistemas digitais.

# Aula 1:

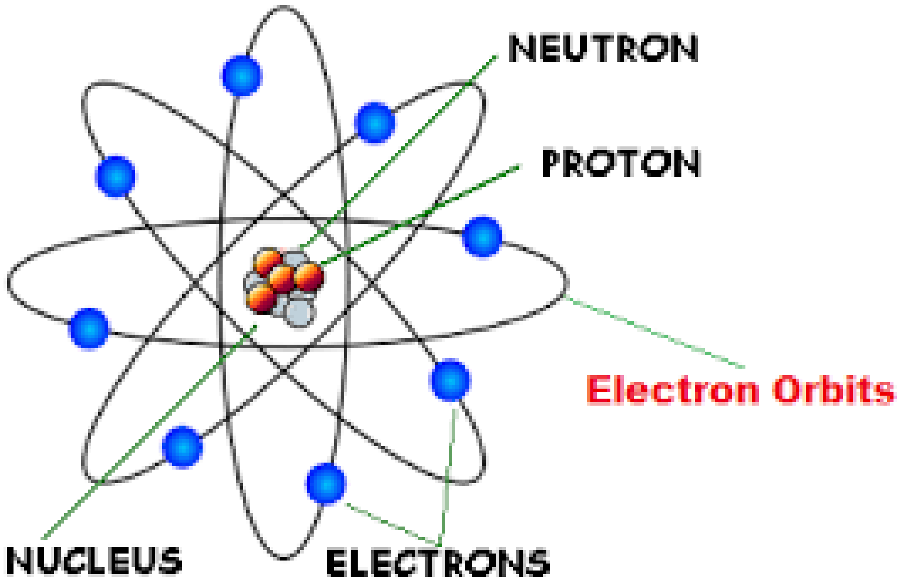
# Fundamentos de circuitos elétricos

# Lógica Digital

## O que é corrente elétrica?

Tudo é feito de átomos. No núcleo de cada átomo, existem dois tipos de partículas: nêutrons, que apresentam carga neutra, e prótons, que apresentam carga positiva.

Ao redor do átomo, na forma de orbitais, existem os elétrons, que apresentam carga negativa. Elétrons e prótons se atraem, ao passo que prótons e prótons se repelem, bem como elétrons e elétrons.



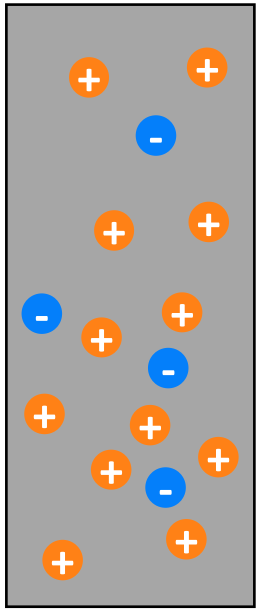

Elétrons tem carga de $-1{,}602 \times 10^{-19}\ C$ (columbs).

# Lógica Digital

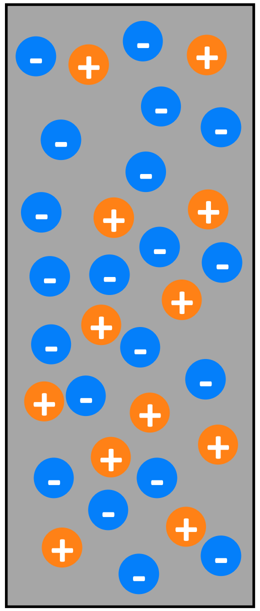
## O que é corrente elétrica?

Alguns dispositivos (como baterias e fontes CCs) possuem dois polos: um deles tem excesso de elétrons, e o outro tem falta de elétrons.

Elétrons e prótons tendem a se atrair, mas essa configuração não permite que isso ocorra.



Polo positivo (+)



Polo negativo (-)

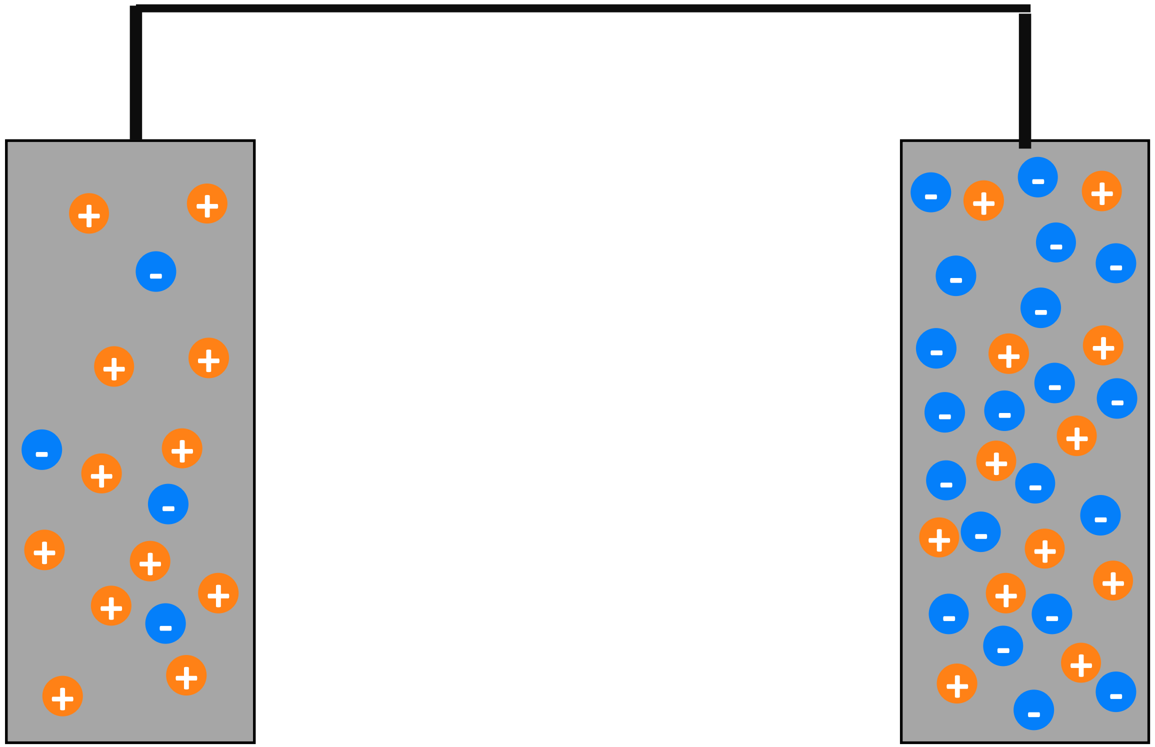
A essa capacidade de atrair outras cargas chamamos de potencial elétrico.

Quando comparamos essa capacidade em dois terminais, a chamamos de diferença de potencial, a qual é medida em volts ($V$).

# Lógica Digital

## O que é corrente elétrica?

Quando uma conexão é feita entre os dois polos, os elétrons em excesso no polo negativo vão para o polo positivo.



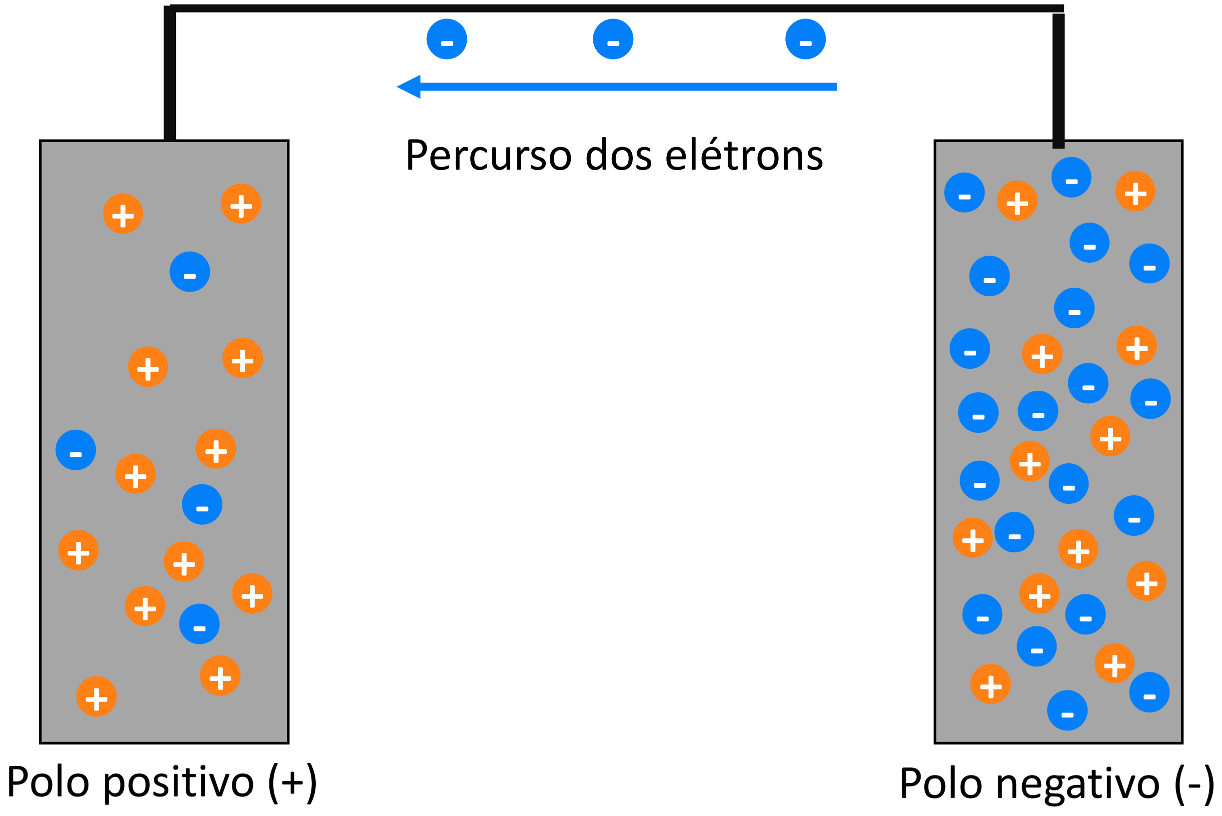
Polo positivo (+)

Polo negativo (-)

# Lógica Digital

## O que é corrente elétrica?

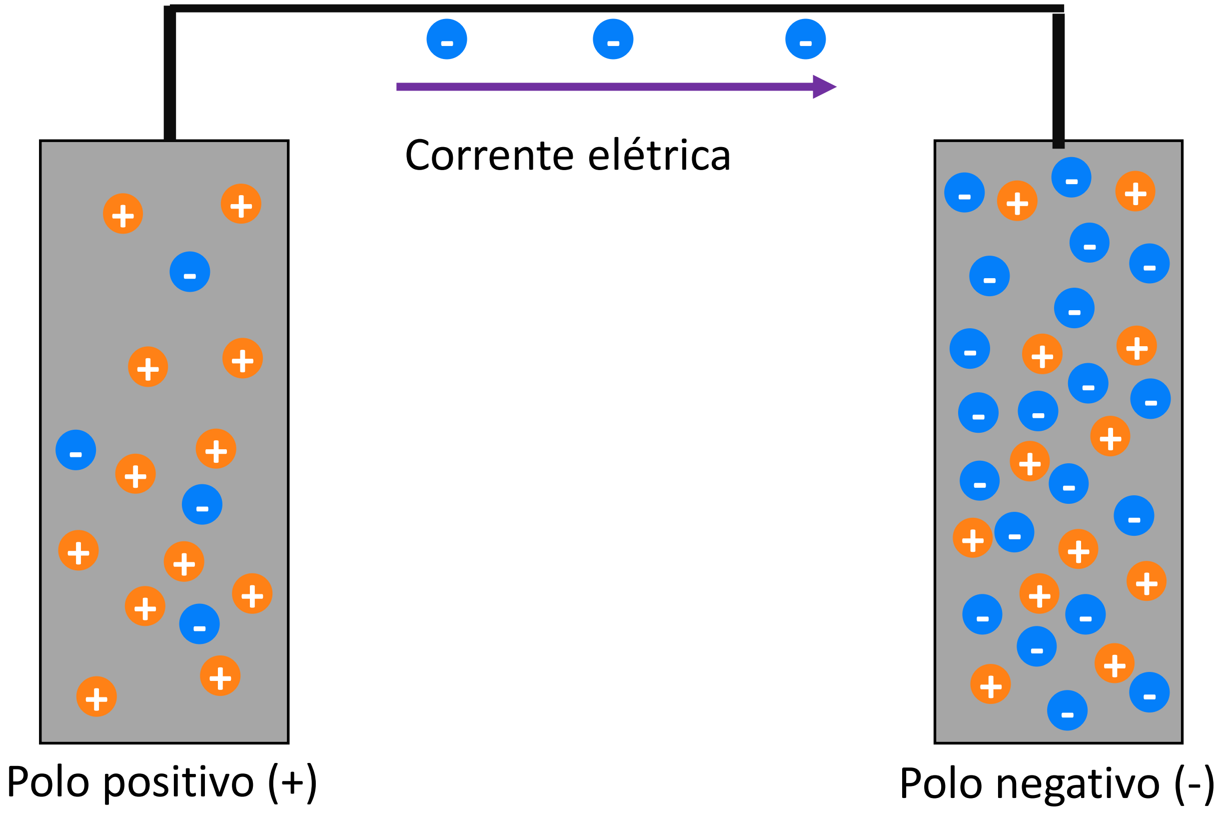
Quando uma conexão é feita entre os dois polos, os elétrons em excesso no polo negativo vão para o polo positivo.



# Lógica Digital

## O que é corrente elétrica?

Chamamos de corrente elétrica **o percurso inverso do percorrido pelos elétrons**.




A unidade da corrente elétrica é o ampere $(A)$, que corresponde a columbs por segundo.

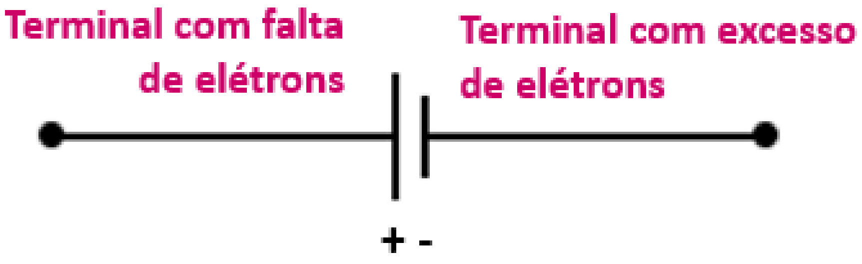
$$[A] = \frac{[C]}{[S]}$$

# Lógica Digital

## Quais componentes fornecem diferença de potencial?

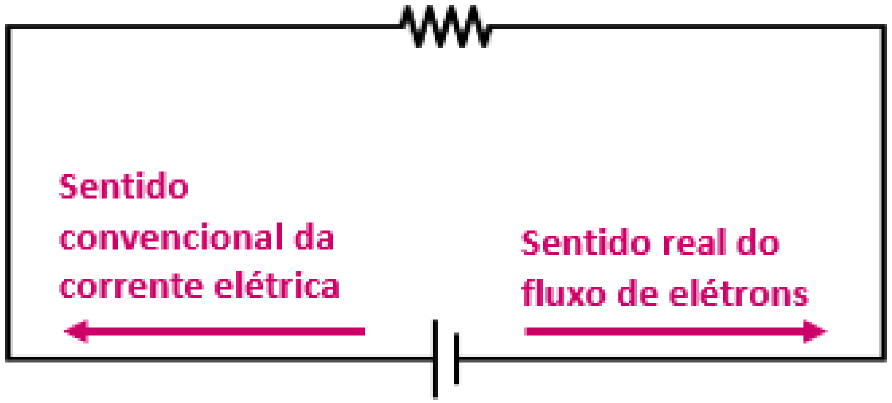
Baterias, pilhas e fontes.











# Lógica Digital

## O que é resistência elétrica?

Ao definirmos um caminho para os elétrons percorrem, estamos lidando com corpos que permitem a passagem desses elétrons.

A **capacidade desse corpo de se opor a passagem de corrente elétrica mesmo quando há uma diferença de potencial aplicada** é chamada de resistência elétrica. Ela é medida em ohms (Ω).

Todo corpo possui resistência elétrica, por menor que ela seja.  Pela primeira Lei de Ohm, a corrente elétrica num corpo é dada por:

$$i = \frac{U}{R}$$

Em que:
$i$: corrente elétrica
$R$: resistência do corpo
$U$: diferença de potencial aplicada

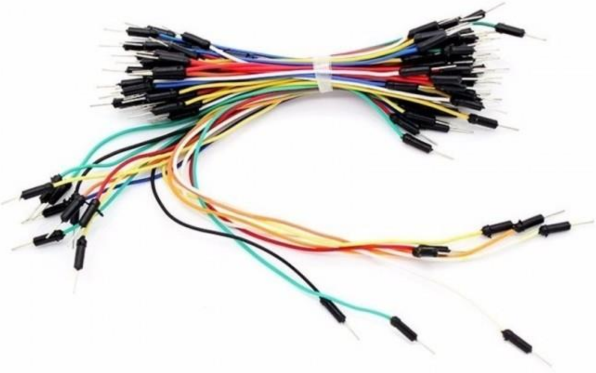
# Lógica Digital

## O que é resistência elétrica?

Fios e LEDs possuem resistência elétrica muito baixa. Isso significa que, ao aplicar uma diferença de potencial neles, a corrente elétrica será muito alta:

$$\lim_{R \to 0} \frac{U}{R} = +\infty$$

Uma corrente muito alta pode danificar esse componente.

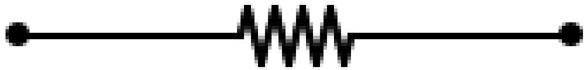



# Lógica Digital

## O que é resistência elétrica?

Para evitar que isso aconteça existem os resistores.

Resistores são componentes que têm por finalidade oferecer uma oposição à passagem de corrente elétrica. Na prática, eles podem ser usados para transformar energia elétrica em outra forma de energia (como térmica) ou para reduzir o valor da corrente em um circuito, garantindo que os demais componentes operem em uma faixa segura.

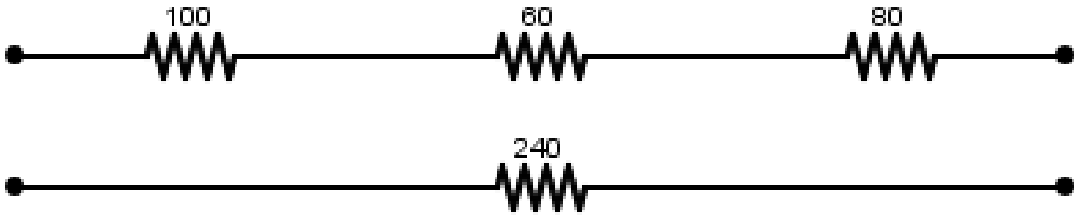


# Lógica Digital

## O que é resistência elétrica?

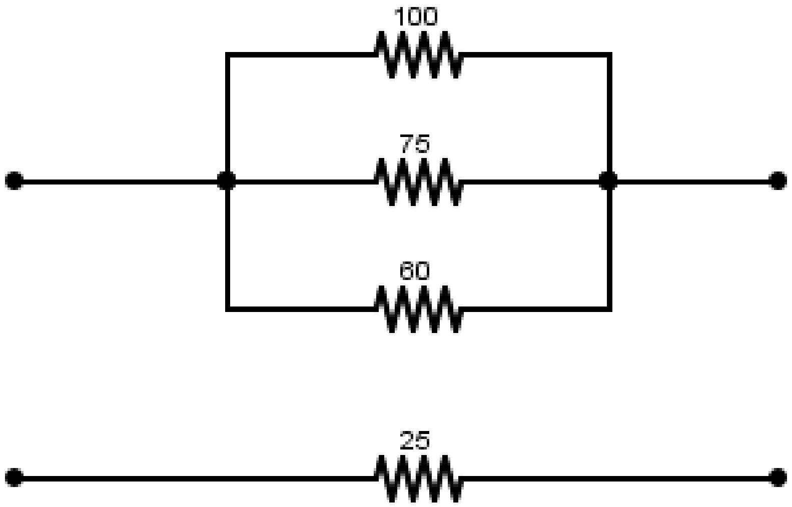
Podemos calcular a resistência equivalente de um conjunto de resistores. Ou seja, encontramos um resistor que tem a mesma resistência que o conjunto.

### Associação em série




$$R_{eq} = \sum_{i=1}^{n} R_i = R_1 + R_2 + \cdots + R_i$$

### Associação em paralelo

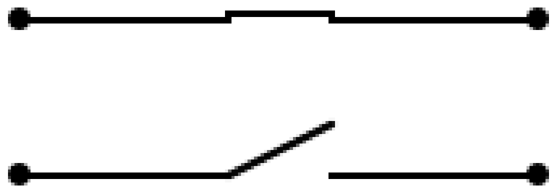



$$\frac{1}{R_{eq}} = \sum_{i=1}^{n} \frac{1}{R_i} = \frac{1}{R_1} + \frac{1}{R_2} + \cdots + 1/R_n$$

# Lógica Digital

## O que é resistência elétrica?

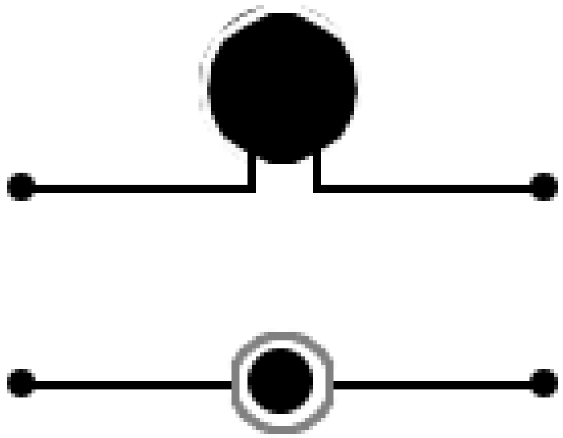
Outro elemento bastante comum nos circuitos é a chave. Ela tem a função de interromper ou permitir a passagem de corrente.



# Lógica Digital

## O que é resistência elétrica?

Visando obter um indicador visual dá passagem de corrente por uma dada região, podemos conectar uma lâmpada ou um LED naquele trecho do circuito. Esses componentes agem como um tipo particular de resistor, convertendo energia elétrica em energia luminosa



# Aula 2:

# Fundamentos de circuitos elétricos

*(prático)*

# Aula 3:

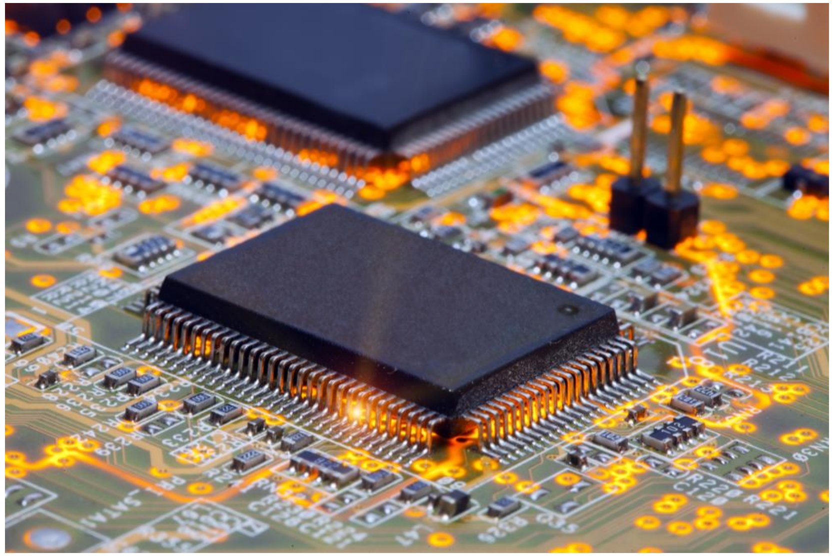
# O transistor

# Lógica Digital

## O que são semicondutores?

Semicondutores são sólidos geralmente cristalinos cuja condutividade elétrica é sensível a condições ambientais, como temperatura ou estado elétrico.

Assim, um semicondutor pode ser isolante ou condutor, conforme as condições ambientais.



# Lógica Digital

## O que são semicondutores?

Um semicondutor pode ser intrínseco ou dopado. Os semicondutores dopados contêm cerca de 1 átomo de um elemento químico indesejado (portanto, uma impureza) para cada bilhão de átomos do material do semicondutor.

Já os semicondutores dopados possuem, intencionalmente, cerca de 1 átomo de um elemento químico desejado (chamado de dopante) para cada milhão de átomos do material do semicondutor

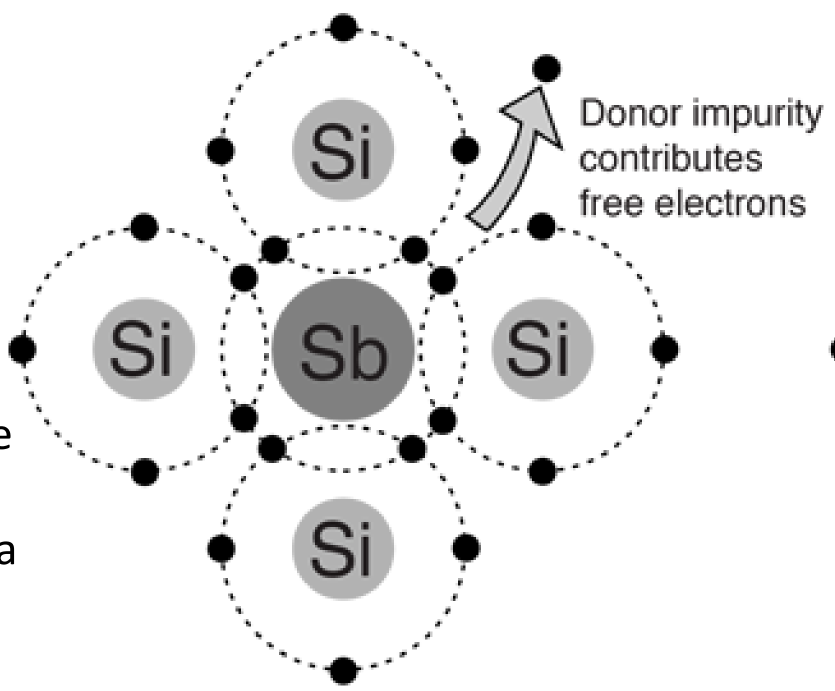
# Lógica Digital

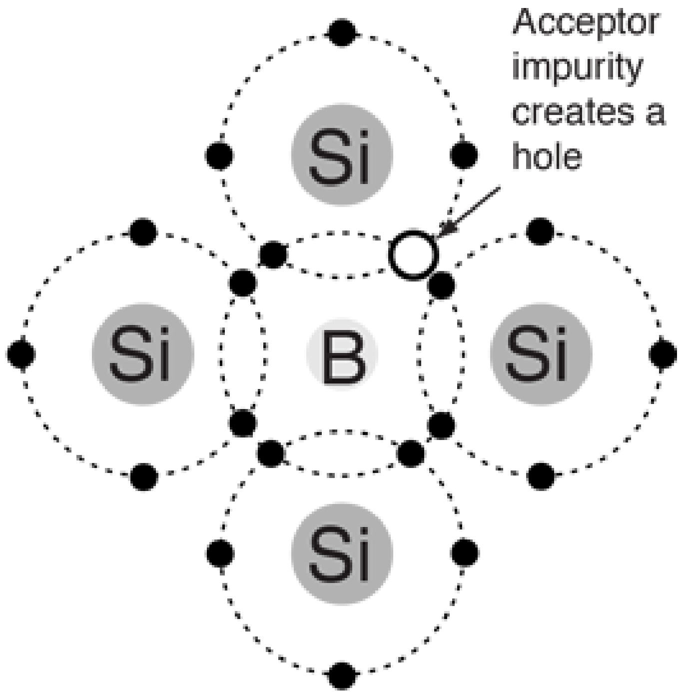
## O que são semicondutores?

Existem dois tipos de semicondutores dopados: do tipo N e do tipo P.

Nos semicondutores do tipo N, ocorre a adição de materiais que, ao realizar as ligações químicas com o material do semicondutor, ficam com elétrons livres. Portanto, o semicondutor do tipo N apresenta carga negativa.

Já nos semicondutores do tipo P, ocorre a adição de materiais que, ao realizar ligações químicas com o material do semicondutor, criam lacunas (por possuírem menos elétrons que a quantidade de ligações realizada). Por possuir elétrons a menos, o semicondutor do tipo P apresenta carga positiva.0



# Lógica Digital

## O que é um transistor?

Existe um componente capaz de controlar a passagem de corrente elétrica por meio de um sinal elétrico: o transistor.

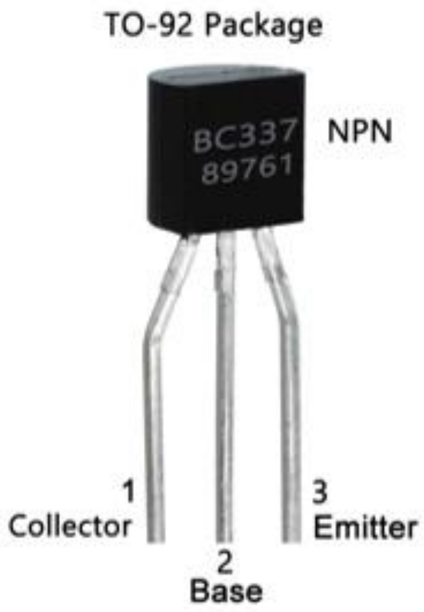
Transistores são componentes feitos de material semicondutor que possuem duas funções: a de amplificação (que não é tão importante em nosso contexto) e a de controle de corrente, fundamental para o entendimento de circuitos digitais.

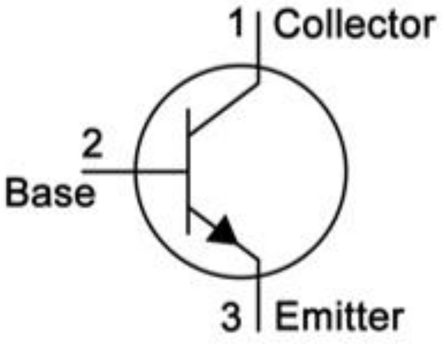
# Lógica Digital

## O que é um transistor?

Transistores podem ser entendidos como pequenas chaves, mas que, ao invés de serem acionadas por um impulso mecânico, são acionadas por um sinal elétrico.

Existem diversos tipos de transistores, mas, aqui, abordaremos um tipo particular: os transistores NPN BC337.



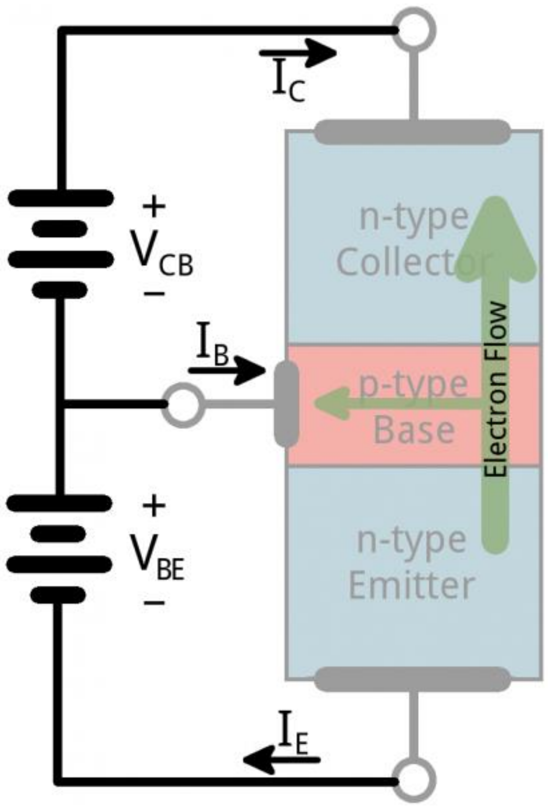

Esses transistores (assim como praticamente todos os outros) possuem três terminais: o terminal coletor, base e emissor.

Cada um desses terminais corresponde a um pino do componente.

# Lógica Digital

## O que é um transistor?

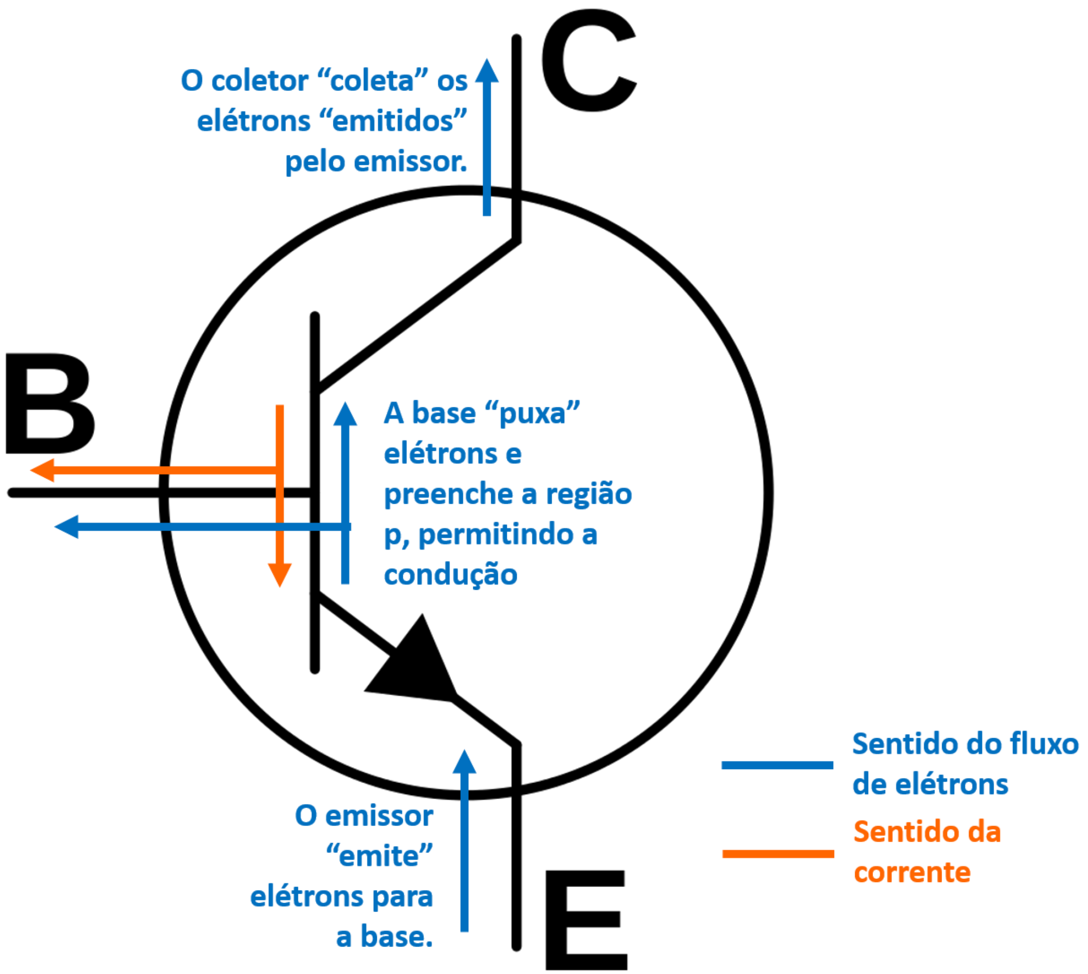
Internamente, o transistor NPN é constituído de três regiões: duas de um semicondutor com dopagem N, e uma de um semicondutor com dopagem P, conforme o esquema:




Normalmente, elétrons podem fluir com facilidade de uma região N para uma região P, mas fluir de uma região P para uma região N requer muita voltagem. Em um semicondutor, elétrons podem facilmente fluir da região P para a região N se a tensão na base for maior que a tensão no emissor.

# Lógica Digital

## O que é um transistor?




O transistor é projetado para passar elétrons do emissor para o coletor (então a corrente flui do coletor para o emissor). O emissor "emite" elétrons para a base, que controla o número de elétrons que o emissor "emite". A maior parte dos elétrons é "coletada" pelo coletor, que os envia para a próxima parte do circuito

O funcionamento de um transistor é bastante complexo. Porém, para nossas aplicações, consideraremos que todo transistor NPN (o único aqui utilizado) funciona da seguinte forma: se há tensão no pino base, então há passagem de corrente elétrica do coletor para o emissor. Caso contrário, não há passagem de corrente

# Lógica Digital

## O que é um transistor?

Na prática, nossos transistores terão, em sua configuração mais simples, o emissor ligado a um fio doador de elétrons (terra ou carregado negativamente). O coletor e a base ficarão ligados a uma fonte de alimentação, ambas de um valor fixo de $5\ \mathrm{V}$.

A base, por sua vez, também estará ligada a uma chave manual que controla a entrada do transistor, mas que pode ser substituída pela saída de outro transistor se assim necessário. Haverá uma resistência entre a chave e a base, a qual deve ter um valor entre $4000$ e $20000\ \Omega$.

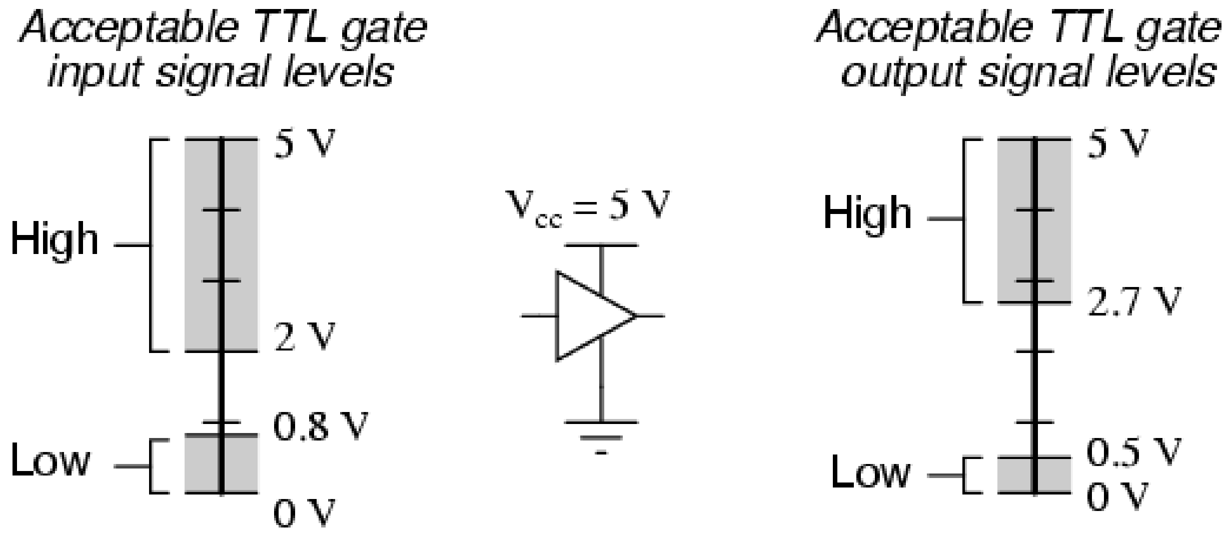
Em algum ponto entre o fio terra do emissor e a fonte do coletor também haverá uma resistência, de valor variável conforme o propósito do circuito. Em algum ponto desse percurso, também haverá um fio de saída, que pode ser ligado a um multímetro ou a um LED, ou, ainda, a outro circuito como entrada do transistor.

# Lógica Digital

## Quantos volts é 1, e quantos volts é 0?

Já sabemos por quais razões este curso é chamado de lógica digital: existe um número finito de estados que um sinal pode exibir. Mais especificamente, existem apenas dois sinais: 1 e 0, alta ou baixa voltagem, respectivamente.

Idealmente, a alta voltagem corresponde a tensão da fonte de alimentação, que no nosso caso é 5 V. Também idealmente, a baixa voltagem corresponde a tensão nula, que no nosso caso é 0 V.

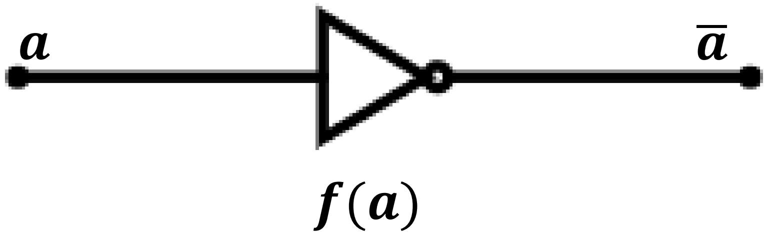


# Aula 4:

# Portas lógicas

# Lógica Digital

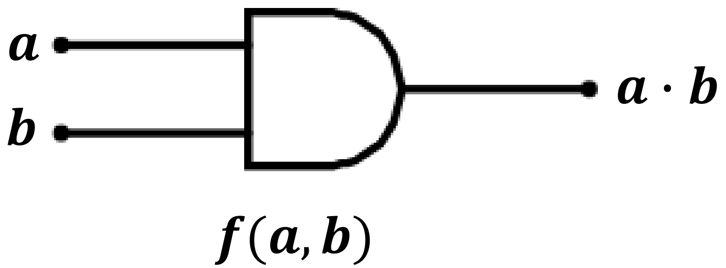
**Podemos representar as portas lógicas por expressões matemáticas!**




| $a$ | $f(a) = \bar{a}$ |
|---|---|
| 0 | 1 |
| 1 | 0 |

# Lógica Digital

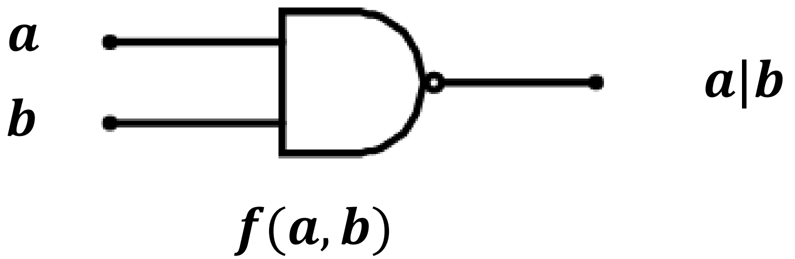
**Podemos representar as portas lógicas por expressões matemáticas!**




| $a$ | $b$ | $f(\mathrm{a}, \mathrm{b}) = a \cdot b$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

# Lógica Digital

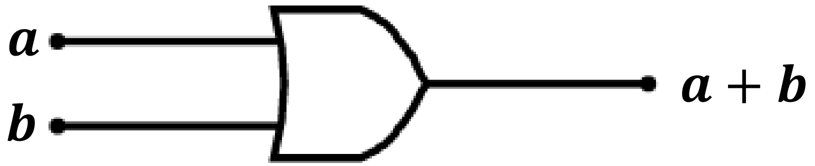
**Podemos representar as portas lógicas por expressões matemáticas!**




| $a$ | $b$ | $f(\mathbf{a},\mathbf{b}) = a\|b$ |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

# Lógica Digital

**Podemos representar as portas lógicas por expressões matemáticas!**

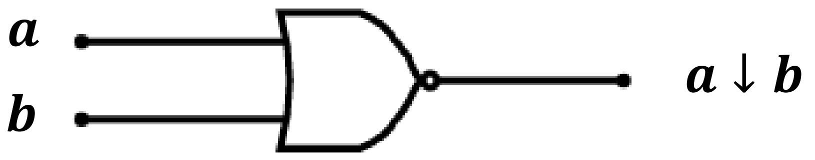



$f(a, b)$

| $a$ | $b$ | $f(\mathrm{a}, \mathrm{b}) = a + b$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

# Lógica Digital

**Podemos representar as portas lógicas por expressões matemáticas!**

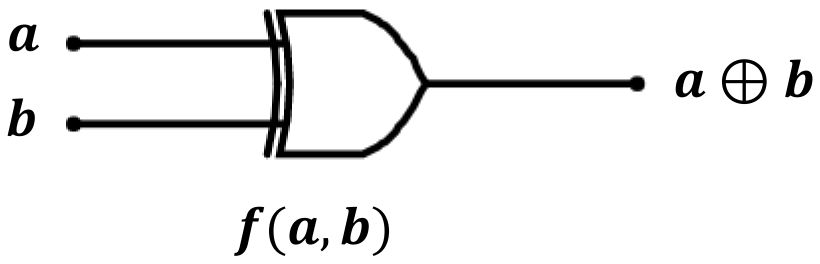



$f(a, b)$

| $a$ | $b$ | $f(\mathrm{a}, \mathrm{b}) = a \downarrow b$ |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

# Lógica Digital

**Podemos representar as portas lógicas por expressões matemáticas!**




| $a$ | $b$ | $f(\mathrm{a}, \mathrm{b}) = a \oplus b$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

# Aula 5:

# Portas lógicas com transistor

*(prático)*

# Aula 6:

# Circuitos integrados

*(prático)*

# Lógica Digital

## O problema com transistores

Vimos que até mesmo uma operação simples (como o XOR) envolve um circuito bastante complexo usando transistores.

Então como montar tais circuitos complexos, na prática?

# Lógica Digital

## O problema com transistores

Vimos que até mesmo uma operação simples (como o XOR) envolve um circuito bastante complexo usando transistores.

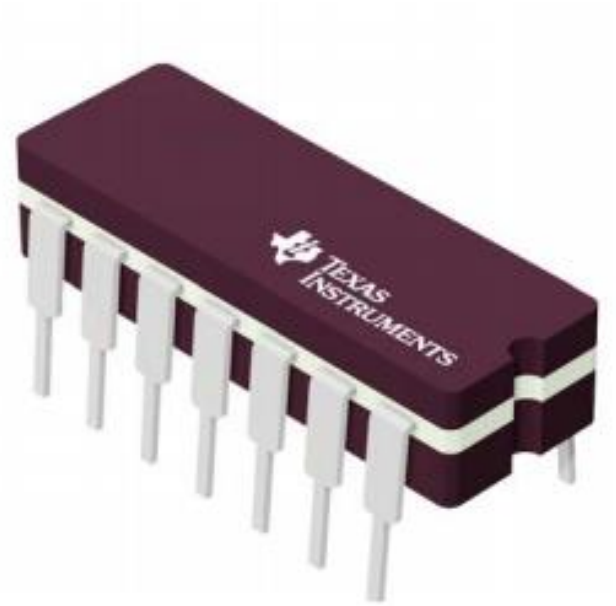
Então como montar tais circuitos complexos, na prática?

## Circuitos integrados

Circuitos integrados são chips que podem conter milhares de transistores, resistores e capacitores em uma dimensão muito diminuta.

A série 7400 de circuitos integrados contém, entre muitas outras coisas, seis chips com as portas lógicas fundamentais.

Hoje, veremos como utilizá-los para testar o funcionamento de portas lógicas simples.

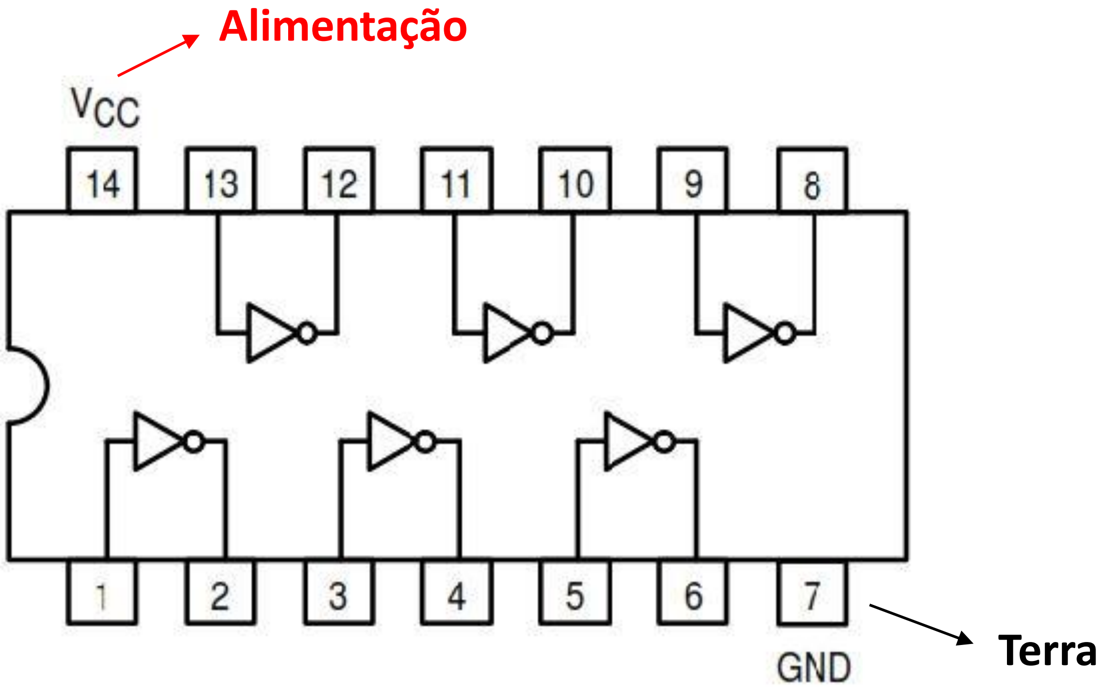


# Lógica Digital

## Alimentação, terra e valores-verdade

Todo circuito possui um pino marcado como GND e outro marcado como VCC.

O pino GND deve receber a referência de 0V, e o pino VCC deve estar ligado à alimentação (5V na maioria dos casos).




# Lógica Digital

## Alimentação, terra e valores-verdade

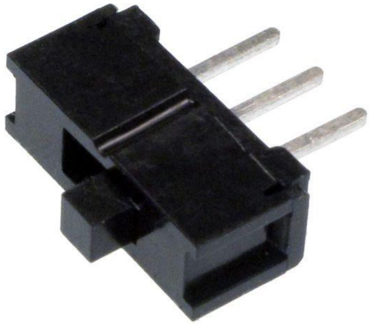
Todo circuito possui um pino marcado como GND e outro marcado como VCC.

O pino GND deve receber a referência de 0V, e o pino VCC deve estar ligado à alimentação (5V na maioria dos casos).

Ao fornecer um bit para algum dos pinos, devemos fornecer a alimentação (caso 1) ou o terra (caso 0).

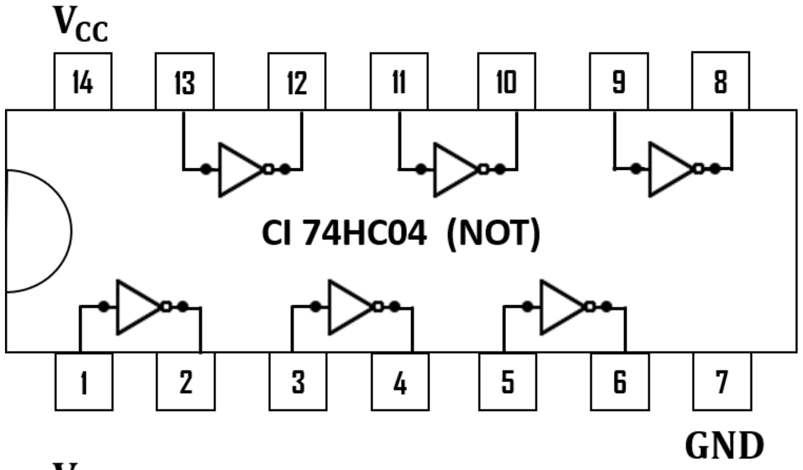
Jamais fornecer 0 usando uma chave táctil comum. Sempre use uma deslizante, ou “espete” os conectores na trilha de alimentação ou do terra.

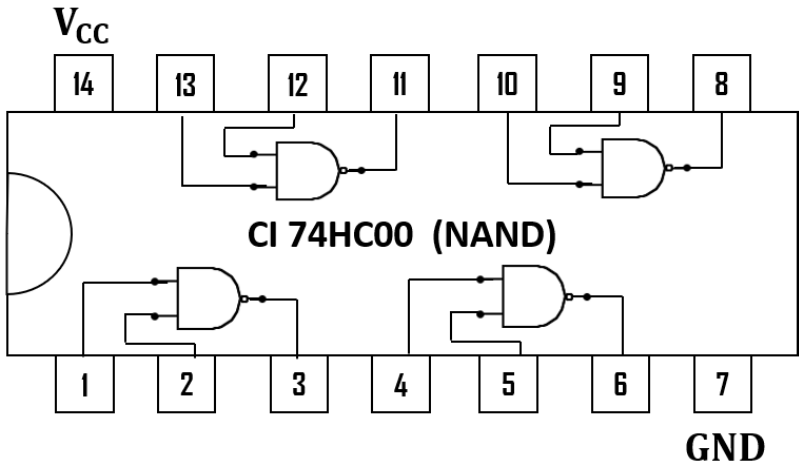
Para a saída, use um LED **em série** com um resistor.

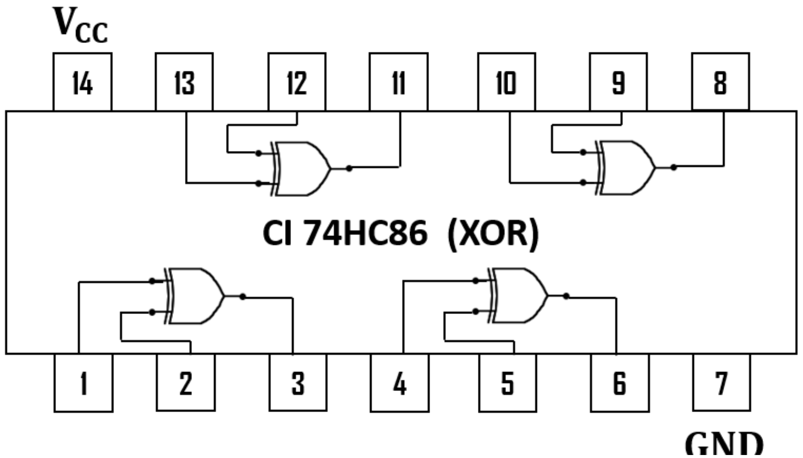


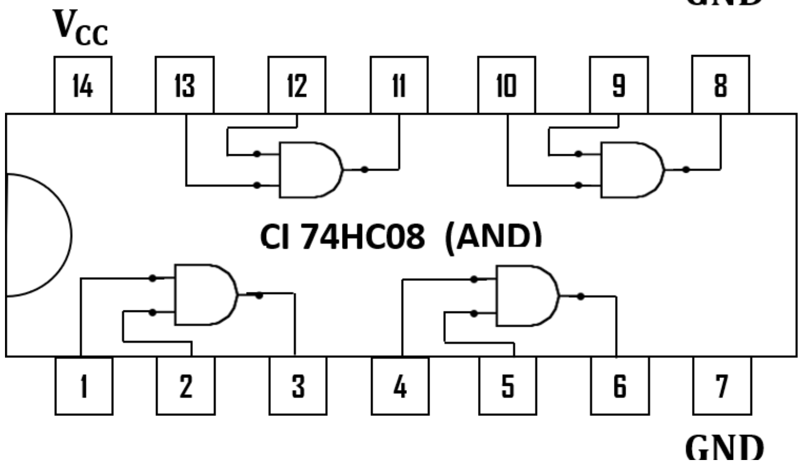
# Lógica Digital

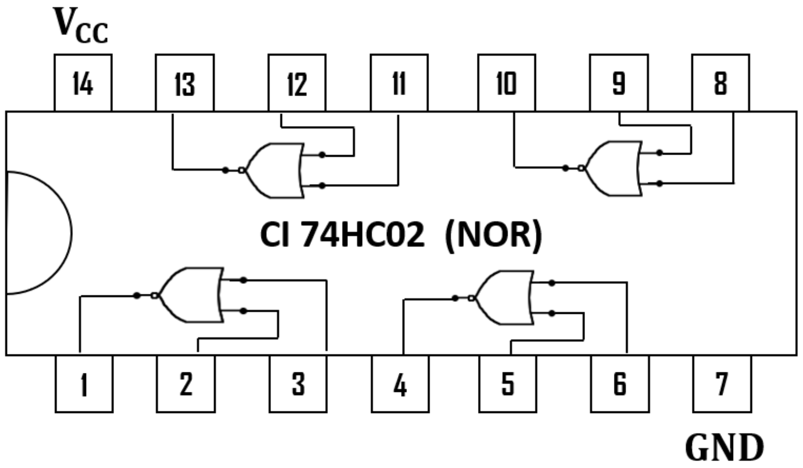
## Circuitos integrados

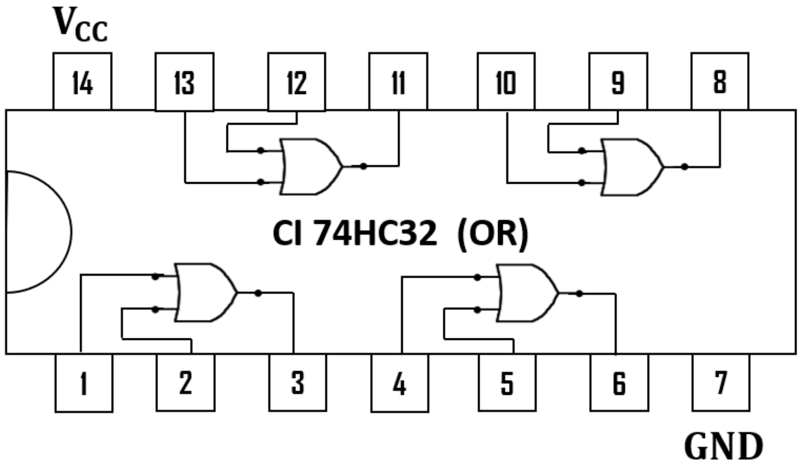


# Aula 7:

# Descrevendo circuitos lógicos

# Lógica Digital

## Motivação

Já sabemos quais são os blocos fundamentais de todos os circuitos e sabemos bem como eles funcionam. É chegado, então, o momento de associarmos estes blocos em circuitos maiores e de aprendermos como um circuito e seu comportamento podem ser representados.

# Lógica Digital

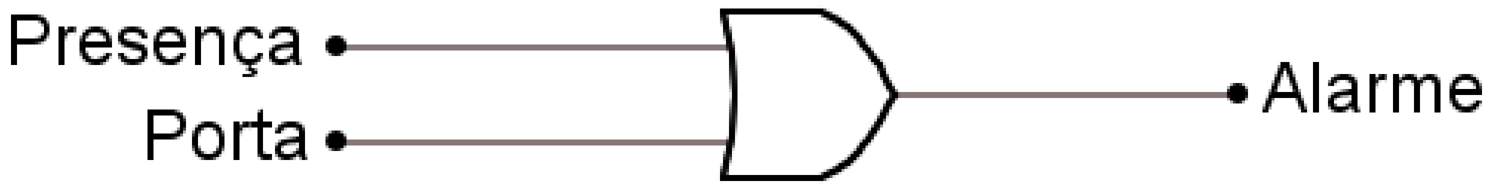
## Projetando um alarme

*“Um alarme para um consultório médico deve ser ativado sempre que uma porta for aberta, ou que um sensor de presença detecte alguém na recepção. Construa um circuito capaz de ativar o alarme com um sinal 1, tendo a disposição um sensor de presença que devolve 1 sempre que detectar algo a frente, e um sensor na porta que devolve 1 sempre que a porta for aberta”*

# Lógica Digital

## Projetando um alarme

*“Um alarme para um consultório médico deve ser ativado sempre que uma porta for aberta, ou que um sensor de presença detecte alguém na recepção. Construa um circuito capaz de ativar o alarme com um sinal 1, tendo a disposição um sensor de presença que devolve 1 sempre que detectar algo a frente, e um sensor na porta que devolve 1 sempre que a porta for aberta”*



# Lógica Digital

## Projetando um aquecedor

***“Um aquecedor deve ligar sempre que a temperatura de uma sala estiver abaixo de $9°\ C$ e quando a hora no momento estiver entre 6 e 7 da manhã. Construa um circuito com uma saída, que deve ser 1 para ligar o aquecedor e 0 para desliga-lo, tendo a disposição um sensor de temperatura que devolve 1 caso a temperatura seja maior ou igual a $9°\ C$ e 0 caso contrário, e um relógio que devolve 1 sempre que a hora estiver entre 06:00 e 07:00.”***

# Lógica Digital

## Projetando um aquecedor

***"Um aquecedor deve ligar sempre que a temperatura de uma sala estiver abaixo de $9°\,C$ e quando a hora no momento estiver entre 6 e 7 da manhã. Construa um circuito com uma saída, que deve ser 1 para ligar o aquecedor e 0 para desliga-lo, tendo a disposição um sensor de temperatura que devolve 1 caso a temperatura seja maior ou igual a $9°\,C$ e 0 caso contrário, e um relógio que devolve 1 sempre que a hora estiver entre 06:00 e 07:00."***

Agora, já temos um problema um pouco mais complicado:

- O sensor de temperatura funciona de forma inversa, devolvendo 1 na situação em que não precisamos ligar o aquecedor.

A solução para esse problema envolve, portanto, o uso de uma porta NOT invertendo a saída do sensor de temperatura, e uma porta AND conjugando a saída da porta NOT e do relógio.

# Lógica Digital

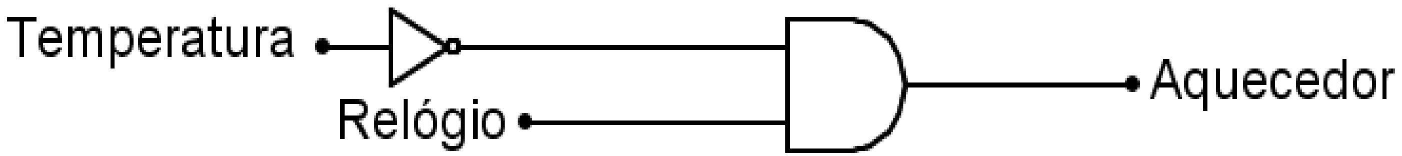
## Projetando um aquecedor

***"Um aquecedor deve ligar sempre que a temperatura de uma sala passar de $9° C$, quando a hora no momento estiver entre 6 e 7 da manhã. Construa um circuito com uma saída, que deve ser 1 para ligar o aquecedor e 0 para desliga-lo, tendo a disposição um sensor de temperatura que devolve 1 caso a temperatura seja maior ou igual a $9° C$ e 0 caso contrário, e um relógio que devolve 1 sempre que a hora estiver entre 06:00 e 07:00."***

Agora, já temos um problema um pouco mais complicado:

- O sensor de temperatura funciona de forma inversa, devolvendo 1 na situação em que não precisamos ligar o aquecedor.

A solução para esse problema envolve, portanto, o uso de uma porta NOT invertendo a saída do sensor de temperatura, e uma porta AND conjugando a saída da porta NOT e do relógio.



# Lógica Digital

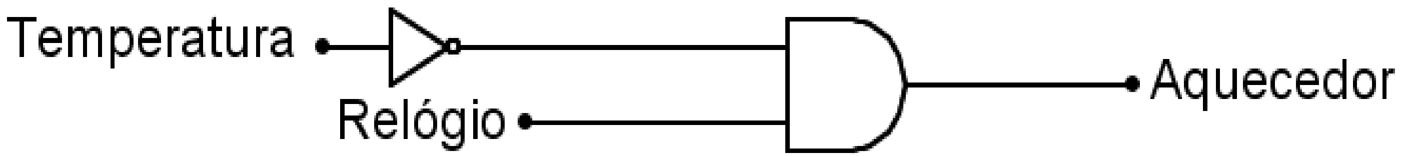
## Projetando um aquecedor

***"Um aquecedor deve ligar sempre que a temperatura de uma sala passar de $9°\ C$, quando a hora no momento estiver entre 6 e 7 da manhã. Construa um circuito com uma saída, que deve ser 1 para ligar o aquecedor e 0 para desliga-lo, tendo a disposição um sensor de temperatura que devolve 1 caso a temperatura seja maior ou igual a $9°\ C$ e 0 caso contrário, e um relógio que devolve 1 sempre que a hora estiver entre 06:00 e 07:00."***

Agora, já temos um problema um pouco mais complicado:

- O sensor de temperatura funciona de forma inversa, devolvendo 1 na situação em que não precisamos ligar o aquecedor.

A solução para esse problema envolve, portanto, o uso de uma porta NOT invertendo a saída do sensor de temperatura, e uma porta AND conjugando a saída da porta NOT e do relógio.



# Lógica Digital

## Projetando um sistema de farol

***"Um sistema de farol com acionamento automático prevê que o farol de um veículo seja acionado sempre que um sensor de luminosidade detectar baixa iluminação no ambiente (representada por uma saída 0) ou sempre que a velocidade do veículo ultrapassar os 80 km/h (representada por uma saída 1 em um sensor de velocidade). Caso o motorista desejar, ele poderá ligar o farol a qualquer momento. Caso também for de seu agrado, ele poderá desativar esse sistema a qualquer momento, ficando apenas com o controle manual".***

Aqui, a situação é um pouco mais complicada para ser resolvida por uma abordagem meramente intuitiva.

# Lógica Digital

## Projetando um sistema de farol

Aqui, a situação é um pouco mais complicada para ser resolvida por uma abordagem meramente intuitiva.

Temos duas situações que podem motivar o acionamento automático do farol: a velocidade acima do limite determinado e a luminosidade abaixo do limite determinado. Com isso, já sabemos que o sensor de luminosidade deve ser ligado a uma porta NOT, e que essa porta NOT, bem como o velocímetro, devem ser ligadas em uma porta OR.

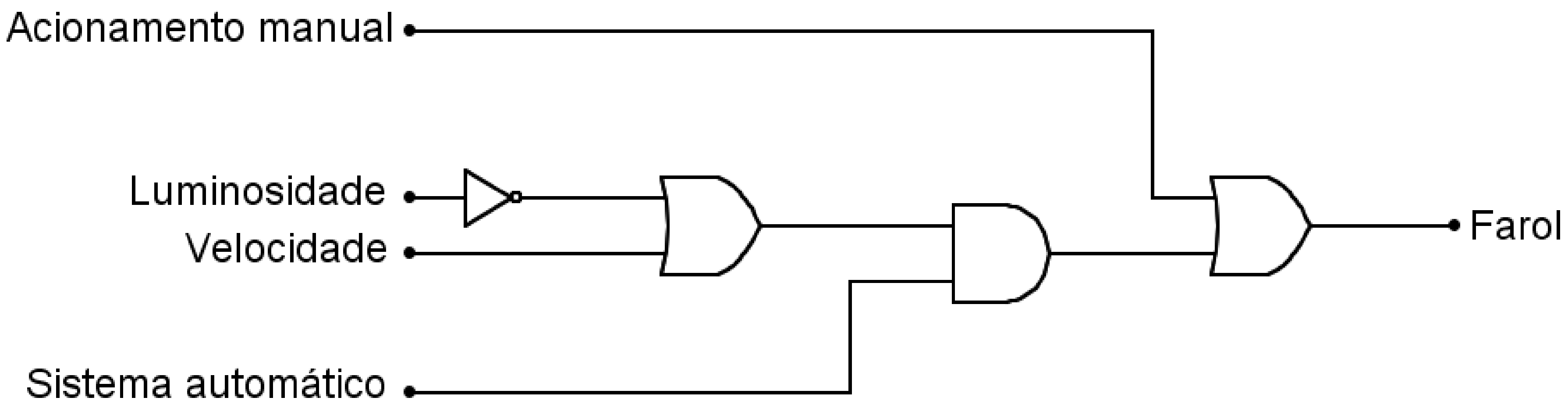
Porém, o sistema automático pode ser desligado a qualquer momento, fazendo com que apenas o sistema manual funcione. Por essa razão, o controle automático também deve estar associado a uma porta AND, que controla a permanência desse sistema.

Sabemos, também, que o farol pode ser acionado a qualquer momento pelo motorista. Portanto, parece razoável associarmos o controle do farol manual com o controle automático em outra porta OR.

# Lógica Digital

## Projetando um sistema de farol

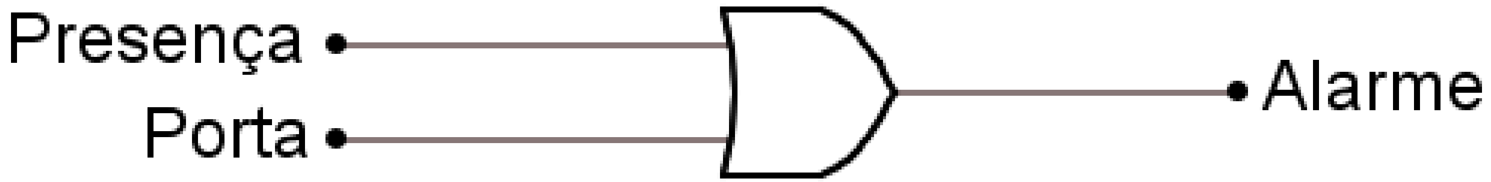
Aqui, a situação é um pouco mais complicada para ser resolvida por uma abordagem meramente intuitiva.



# Lógica Digital

## Expressão booleana

Um circuito pode também ser representado por uma expressão booleana. Veja:




$$alarme(a, b) = a + b$$

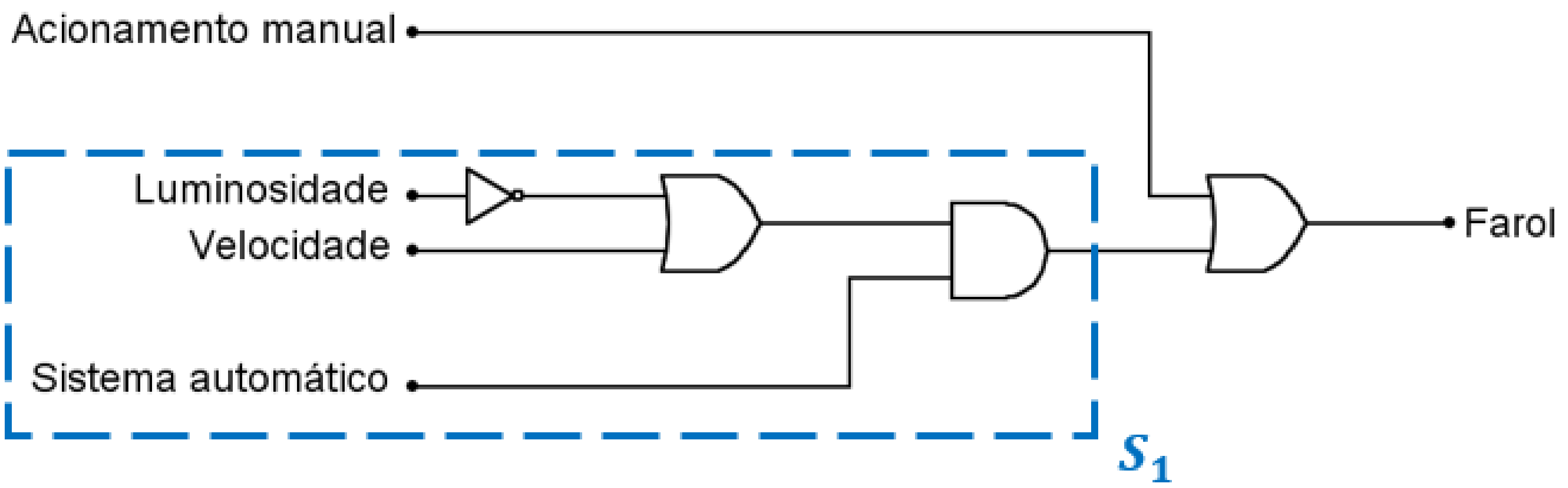
$$alarme(a, b) = a + b$$

$$aquecedor(r, t) = \bar{t} + r$$

# Lógica Digital

## Expressão booleana

Para o circuito do farol, a expressão booleana pode ser obtida por partes.

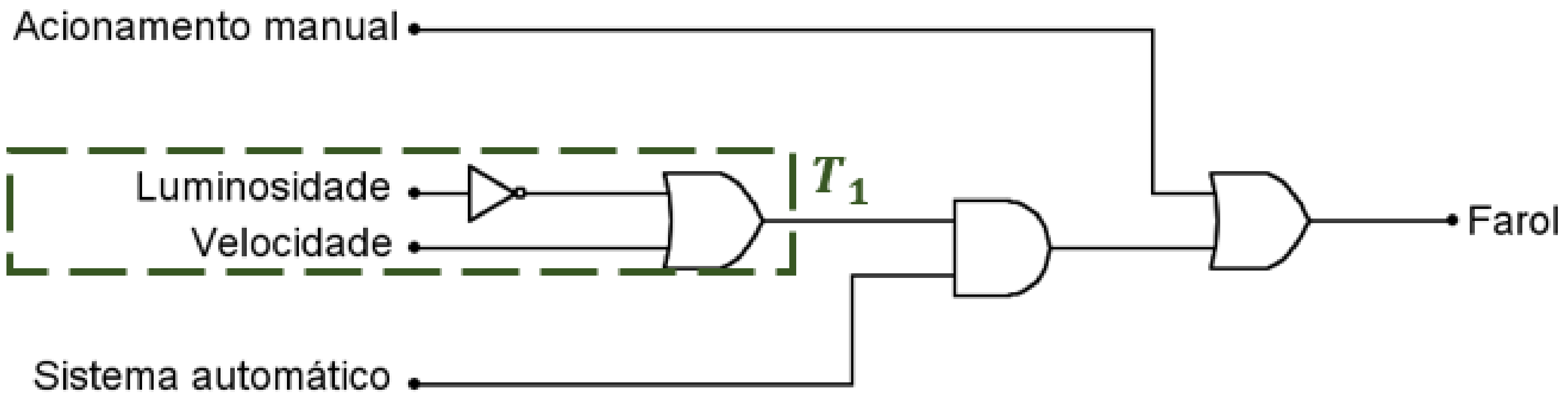



$$f = m + S_1.$$

# Lógica Digital

## Expressão booleana

Para o circuito do farol, a expressão booleana pode ser obtida por partes.



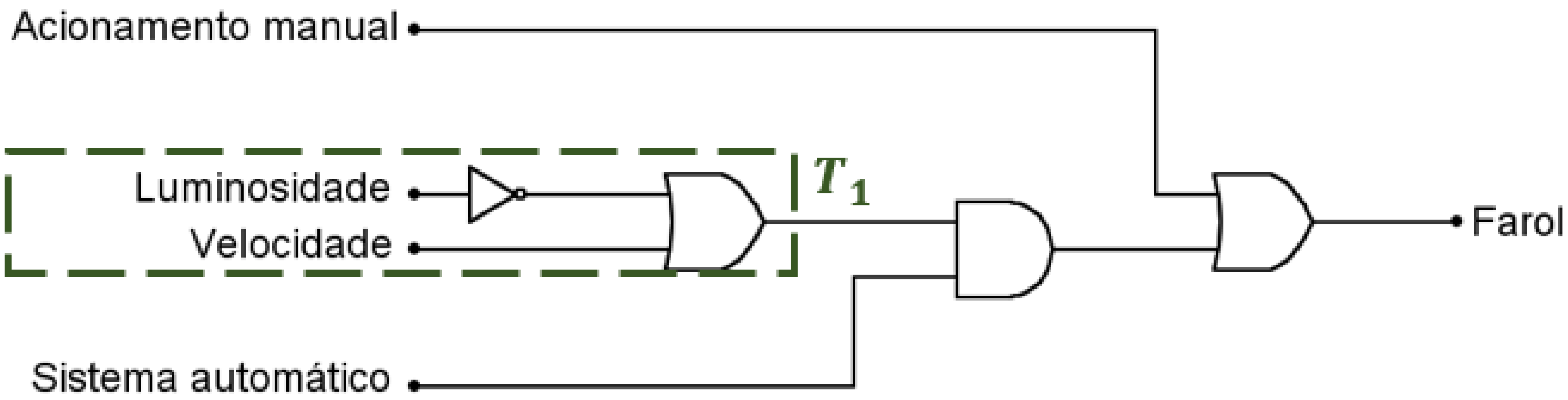

$$S_1 = s \, . \, T_1.$$

$$f = m + S_1.$$

# Lógica Digital

## Expressão booleana

Para o circuito do farol, a expressão booleana pode ser obtida por partes.




$$f = m + S_1.$$

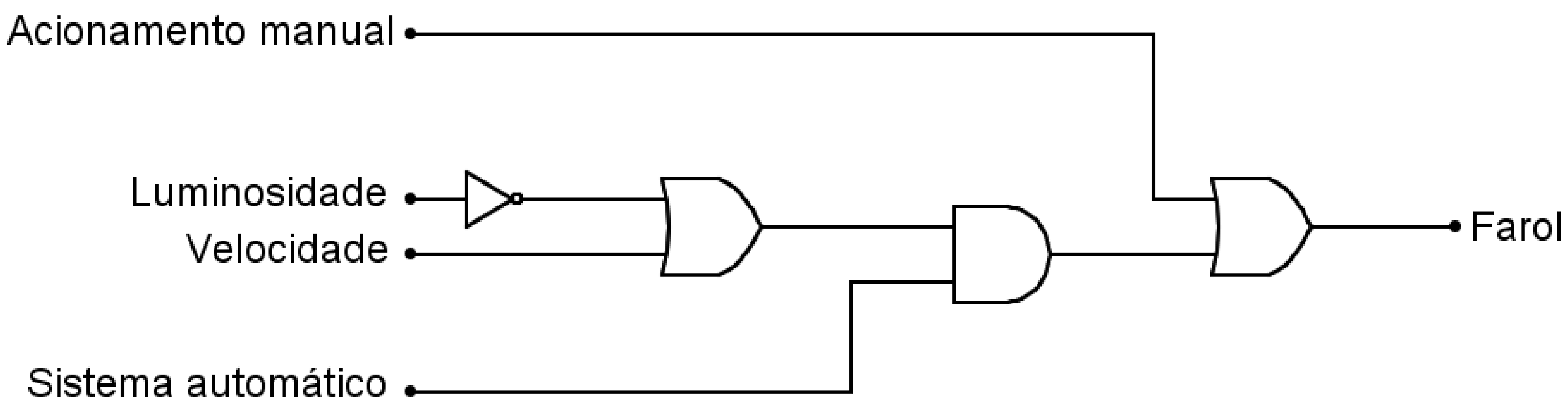
$$S_1 = s \,.\, T_1.$$

$$T_1 = \bar{l} + v$$

# Lógica Digital

## Expressão booleana

Para o circuito do farol, a expressão booleana pode ser obtida por partes.




$$farol(m, l, v, s) = m + \left(s \,.\, (\bar{l} + v)\right)$$

$$f = m + S_1.$$

$$S_1 = s \,.\, T_1.$$

$$T_1 = \bar{l} + v$$

# Aula 8:

# Implementando o circuito do farol na protoboard (*prático*)

# Aula 9:

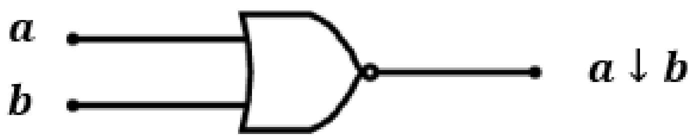
# Tabelas-verdade

# Lógica Digital

## Definição

*Uma tabela-verdade de um circuito é uma tabela que retrata o comportamento do circuito para todas as possibilidades de entrada.*

Já lidamos com tabelas-verdade no passado!




$f(a, b)$

| $a$ | $b$ | $f(\mathrm{a}, \mathrm{b}) = a \downarrow b$ |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

# Lógica Digital

## Fazendo a tabela-verdade

Considere o seguinte circuito:


| $a$ | $b$ | $c$ |
|---|---|---|

Primeiro, adicionamos uma coluna para cada entrada.

# Lógica Digital

## Fazendo a tabela-verdade

Considere o seguinte circuito:


| $a$ | $b$ | $c$ |
|---|---|---|

Quantas possibilidades de entrada ele possui?

# Lógica Digital

## Definição

Considere o seguinte circuito:


| $a$ | $b$ | $c$ |
|---|---|---|

Quantas possibilidades de entrada ele possui?

$$2 \times 2 \times 2 = 8$$

# Lógica Digital

## Definição

Considere o seguinte circuito:


$a \quad b \quad c$

Quantas possibilidades de entrada ele possui?

$$2 \times 2 \times 2 = 8$$

$$2^n$$

# Lógica Digital

## Definição

Considere o seguinte circuito:


Inserimos $2^n$ linhas.

| $a$ | $b$ | $c$ |
|---|---|---|
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |

# Lógica Digital

## Definição

Considere o seguinte circuito:


Preenchemos a linha mais a direita alternando 0 e 1.

| $a$ | $b$ | $c$ |
|---|---|---|
| | | 0 |
| | | 1 |
| | | 0 |
| | | 1 |
| | | 0 |
| | | 1 |
| | | 0 |
| | | 1 |

# Lógica Digital

## Definição

Considere o seguinte circuito:


| $a$ | $b$ | $c$ |
|---|---|---|
| | 0 | 0 |
| | 0 | 1 |
| | 1 | 0 |
| | 1 | 1 |
| | 0 | 0 |
| | 0 | 1 |
| | 1 | 0 |
| | 1 | 1 |

Preenchemos próxima linha alternando 00 e 11.

# Lógica Digital

## Definição

Considere o seguinte circuito:


Preenchemos próxima linha alternando 0000 e 1111.

| $a$ | $b$ | $c$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |
| 1 | 1 | 1 |

# Lógica Digital

## Definição

Considere o seguinte circuito:

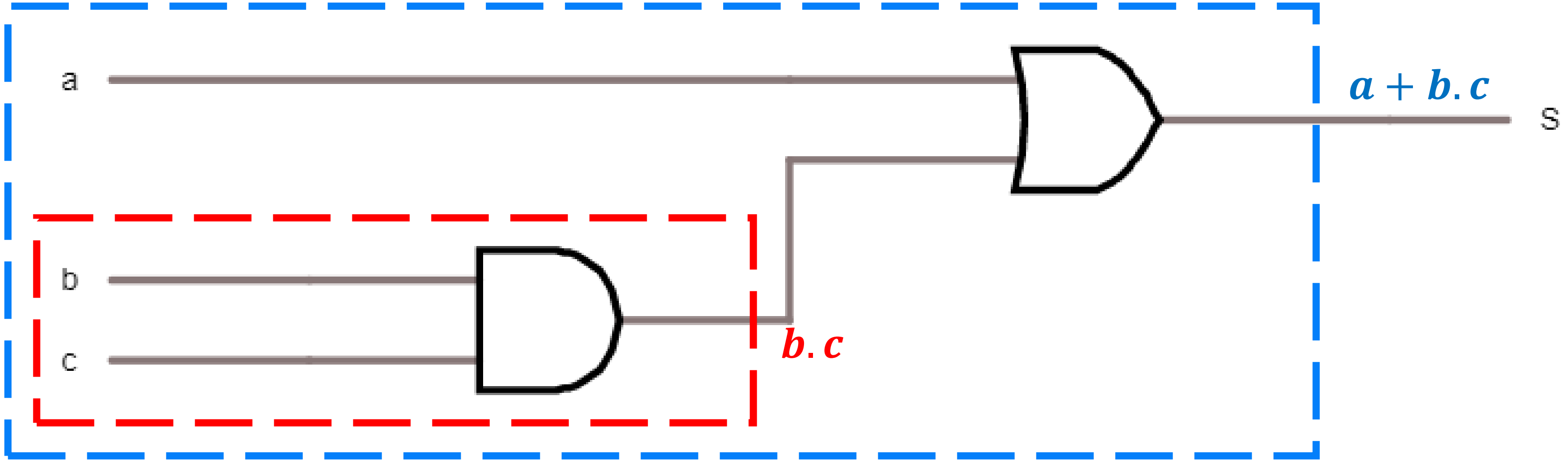

Em cada linha, o número de zeros e uns na repetição é multiplicado por 2.

| $a$ | $b$ | $c$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |
| 1 | 1 | 1 |

# Lógica Digital

## Definição

Considere o seguinte circuito:



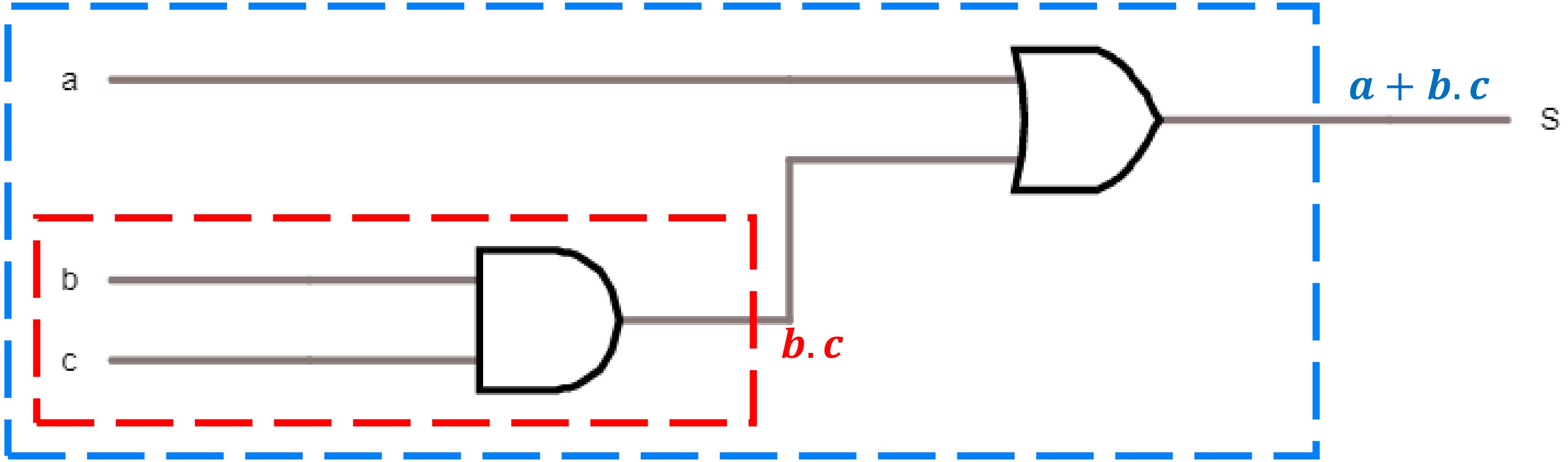

| $a$ | $b$ | $c$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |
| 1 | 1 | 1 |

Em seguida, modularizamos cada parte do circuito.

# Lógica Digital

## Definição

Considere o seguinte circuito:



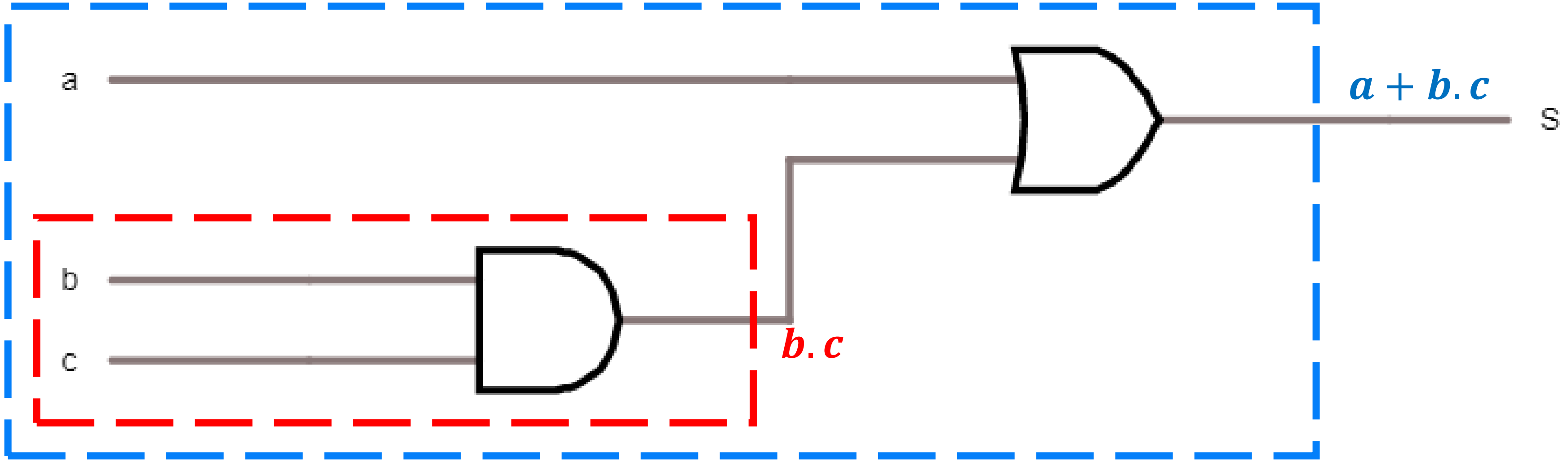

E adicionamos uma coluna para cada parte modularizada.

| $a$ | $b$ | $c$ | $b.c$ | $a+b.c$ |
|---|---|---|---|---|
| 0 | 0 | 0 | | |
| 0 | 0 | 1 | | |
| 0 | 1 | 0 | | |
| 0 | 1 | 1 | | |
| 1 | 0 | 0 | | |
| 1 | 0 | 1 | | |
| 1 | 1 | 0 | | |
| 1 | 1 | 1 | | |

# Lógica Digital

## Definição

Considere o seguinte circuito:



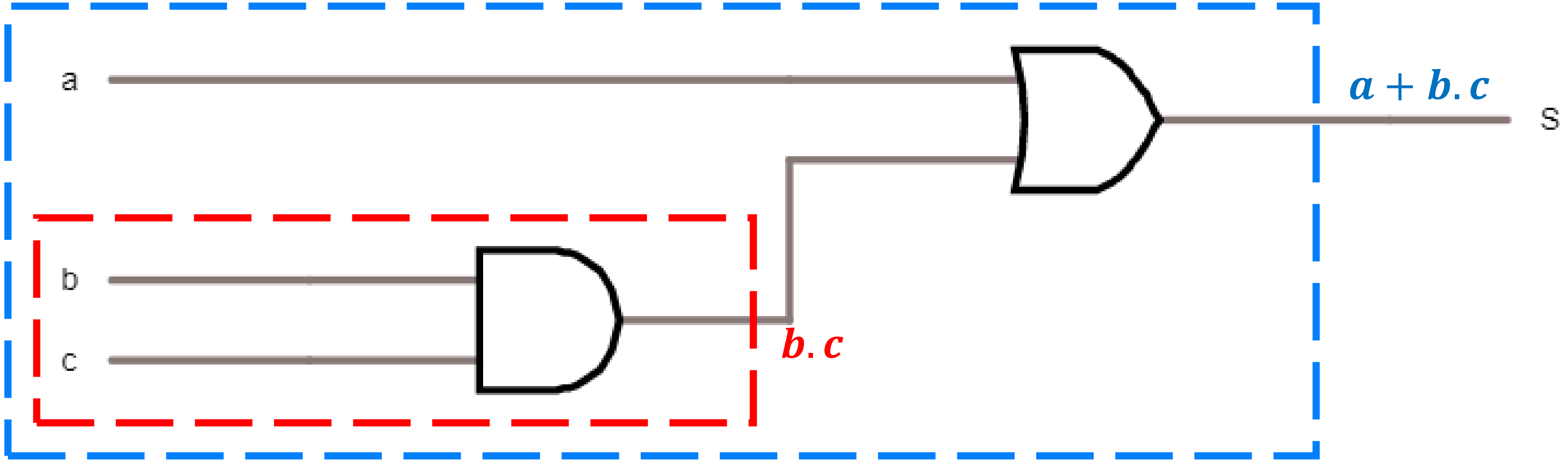

E preenchemos com os valores de cada coluna, até terminar.

| $a$ | $b$ | $c$ | $b.c$ | $a+b.c$ |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | |
| 0 | 0 | 1 | 0 | |
| 0 | 1 | 0 | 0 | |
| 0 | 1 | 1 | 1 | |
| 1 | 0 | 0 | 0 | |
| 1 | 0 | 1 | 0 | |
| 1 | 1 | 0 | 0 | |
| 1 | 1 | 1 | 1 | |

# Lógica Digital

## Definição

Considere o seguinte circuito:




E preenchemos com os valores de cada coluna, até terminar.

| $a$ | $b$ | $c$ | $b.c$ | $a+b.c$ |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

# Aula 10:

# O simulador Falstad (*prático*)

# Aula 11:

# Álgebra de Boole

# Lógica Digital

## Motivação

Sabemos que uma mesma tabela-verdade pode corresponder a vários circuitos que têm o mesmo comportamento. Quanto mais reduzido for um circuito (ou seja, menor número de portas envolvidas), menores serão os custos envolvidos em sua produção, menor será o atraso de propagação e menor será a tendência do circuito a superaquecer. Portanto, é importante possuirmos técnicas que permitam encontrar o menor circuito que corresponde a uma mesma tabela-verdade, denominado circuito mínimo.

# Lógica Digital

## Álgebra de Boole

Uma dessas técnicas envolve a compreensão da álgebra de Boole, um ramo da matemática no qual as variáveis envolvidas são valores-verdade, representados por 1 (verdadeiro) e 0 (falso). Na prática, as portas lógicas aqui estudadas representam operações em álgebra de Boole, e, através do domínio destas operações e de suas propriedades, somos capazes de encontrar expressões equivalentes.

Essas propriedades são dadas na forma de leis. A seguir, enunciaremos todas elas, em que $p$,$q$ e $r$ são variáveis binárias.

# Lógica Digital

## Axiomas

$$\left.\begin{array}{l} 0.0 = 0 \\ 1.1 = 1 \end{array}\right\} \text{Sobre a conjunção}$$

$0.1 = 1.0 = 0$ A ordem dos fatores não importa

$$\left.\begin{array}{l} 1 + 1 = 1 \\ 0 + 0 = 0 \end{array}\right\} \text{Sobre a disjunção}$$

$1 + 0 = 0 + 1 = 1$ A ordem dos fatores não importa

$$\left.\begin{array}{l} x = 0 \leftrightarrow \bar{x} = 1 \\ x = 1 \leftrightarrow \bar{x} = 0 \end{array}\right\} \text{Sobre a negação}$$

# Lógica Digital

## Teoremas de uma variável

$$x.0 = 0$$

$$x.1 = x$$

$$x.x = x$$

$$x.\bar{x} = 0$$

$$\bar{\bar{x}} = x$$

$$x + 1 = 1$$

$$x + 0 = x$$

$$x + x = x$$

$$x + \bar{x} = 1$$

# Lógica Digital

## Teoremas de uma variável

$$x.0 = 0$$

$$x.1 = x$$

$$x.x = x$$

$$x.\bar{x} = 0$$

$$\bar{\bar{x}} = x$$

$$x + 1 = 1$$

$$x + 0 = x$$

$$x + x = x$$

$$x + \bar{x} = 1$$

## Propriedades comutativas

$$x.y = y.x$$
$$x + y = y + x$$

## Propriedades associativas

$$x.(y.z) = (x.y).z$$
$$x + (y + z) = (x + y) + z$$

## Propriedades distributivas

$$x.(y + z) = x.y + x.z$$
$$x + y.z = (x + y).(x + z)$$

# Lógica Digital

## Leis de absorção

$$x + \underbrace{x.y} = x$$

**Ou tem 0, ou tem o mesmo valor de $x$.**

$$x.\underbrace{(x + y)} = x$$

**Ou tem 1, ou tem o mesmo valor de $x$.**

# Lógica Digital

## Leis de absorção

$$x + \underbrace{x.y} = x$$

**Ou tem 0, ou tem o mesmo valor de $x$.**

$$x.\underbrace{(x + y)} = x$$

**Ou tem 1, ou tem o mesmo valor de $x$.**

## Leis de combinação

$$x.y + x.\bar{y} = x$$

$$\left\{\begin{array}{l} x.y + x.\bar{y} = x \\ x(y + \bar{y}) = x \\ x(1) = x \\ x = x \end{array}\right.$$

$$(x + y).(x + \bar{y}) = x$$

$$\left\{\begin{array}{l} x.x + y.x + x.\bar{y} + y.\bar{y} = x \\ x.x + x + 0 = x \\ x + x = x \\ x = x \end{array}\right.$$

# Lógica Digital

## Leis de Morgan

$$\overline{x.y} = \bar{x} + \bar{y}$$

$$\overline{x + y} = \bar{x}.\bar{y}$$

$$x + \bar{x}.y = x + y$$

$$x.(\bar{x} + y) = x.y$$

# Lógica Digital

## Provando a equivalência de expressões lógicas

A primeira possibilidade é fazer as tabelas-verdade para os dois lados da expressão e compará-las.

$$(x_1 + x_3).(\overline{x_1} + \overline{x_3}) = x_1.\overline{x_3} + \overline{x_1}x_3$$

| $x_1$ | $x_3$ | $\overline{x_1}$ | $\overline{x_3}$ | $A = (x_1 + x_3)$ | $B = (\overline{x_1} + \overline{x_3})$ | $A.B$ |
|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 0 | 1 | 0 |
| 1 | 0 | 0 | 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 1 | 0 | 0 |

**Figura 77: tabela-verdade para o lado esquerdo (LHS).**

| $x_1$ | $x_3$ | $\overline{x_1}$ | $\overline{x_3}$ | $A = (x_1.\overline{x_3})$ | $B = \overline{x_1}x_3$ | $A + B$ |
|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 | 0 | 0 | 0 |

**Figura 78: tabela-verdade para o lado direito (RHS).**

# Lógica Digital

## Provando a equivalência de expressões lógicas

Outra possibilidade é através da manipulação algébrica. Aqui, devemos manipular os dois lados ao mesmo tempo, ou somente um dos lados por vez, ou somente um único lado, visando provar a igualdade estabelecida. No nosso exemplo, poderíamos seguir alguns passos: $(x_1 + x_3).(\overline{x_1} + \overline{x_3}) = x_1.\overline{x_3} + \overline{x_1}x_3$

Primeiro, aplicaremos a distributiva no LHS:

$$(x_1 + x_3)(\overline{x_1}) + (x_1 + x_3)(\overline{x_3}) =$$

Aplicando mais uma vez a mesma propriedade:

$$= (x_1\overline{x_1}) + (x_3\overline{x_1}) + (x_1\overline{x_3}) + (x_3\overline{x_3}) =$$

Usando os teoremas:

$$= 0 + (x_3\overline{x_1}) + (x_1\overline{x_3}) + 0 = (x_3\overline{x_1}) + (x_1\overline{x_3}) +$$

# Aula 12:

# Formas canônicas

# Lógica Digital

## Motivação

Veremos agora que, partindo de uma tabela-verdade, existem duas formas canônicas que podem ser obtidas, e que permitem produzir um circuito com o mesmo funcionamento do funcionamento especificado na tabela-verdade.

## Duas formas canônicas

Mintermos e maxtermos são utilizados para reescrever uma função lógica em uma forma padronizada no sentido de obter-se uma simplificação da mesma. Esta simplificação é traduzida na redução do número de portas do circuito lógico que implementa tal função.

# Lógica Digital

## Usaremos uma tabela-verdade pronta de exemplo

Veremos agora que, partindo de uma tabela-verdade, existem duas formas canônicas que podem ser obtidas, e que permitem produzir um circuito com o mesmo funcionamento do funcionamento especificado na tabela-verdade.

| | *a* | *b* | *c* | *S* |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 2 | 0 | 1 | 0 | 0 |
| 3 | 0 | 1 | 1 | 0 |
| 4 | 1 | 0 | 0 | 1 |
| 5 | 1 | 0 | 1 | 1 |
| 6 | 1 | 1 | 0 | 0 |
| 7 | 1 | 1 | 1 | 0 |

**As linhas estão numeradas de 0 a 7. As saídas já estão definidas.**

# Lógica Digital

## Mintermos

Podemos escrever o mintermo de uma linha negando as variáveis que tem valor 0, mantendo as variáveis que tem valor 1, e realizando uma conjunção lógica entre elas.

| | $a$ | $b$ | $c$ | $S$ | |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 | $m_0 = \bar{a}.\bar{b}.\bar{c}$ |
| 1 | 0 | 0 | 1 | 0 | $m_1 = \bar{a}.\bar{b}.c$ |
| 2 | 0 | 1 | 0 | 0 | $m_2 = \bar{a}.b.\bar{c}$ |
| 3 | 0 | 1 | 1 | 0 | $m_3 = \bar{a}.b.c$ |
| 4 | 1 | 0 | 0 | 1 | $m_4 = a.\bar{b}.\bar{c}$ |
| 5 | 1 | 0 | 1 | 1 | $m_5 = a.\bar{b}.c$ |
| 6 | 1 | 1 | 0 | 0 | $m_6 = a.b.\bar{c}$ |
| 7 | 1 | 1 | 1 | 0 | $m_7 = a.b.c$ |

# Lógica Digital

## Soma de mintermos

Para obter essa forma canônica, basta somar as linhas cuja saída da tabela verdade é 1.

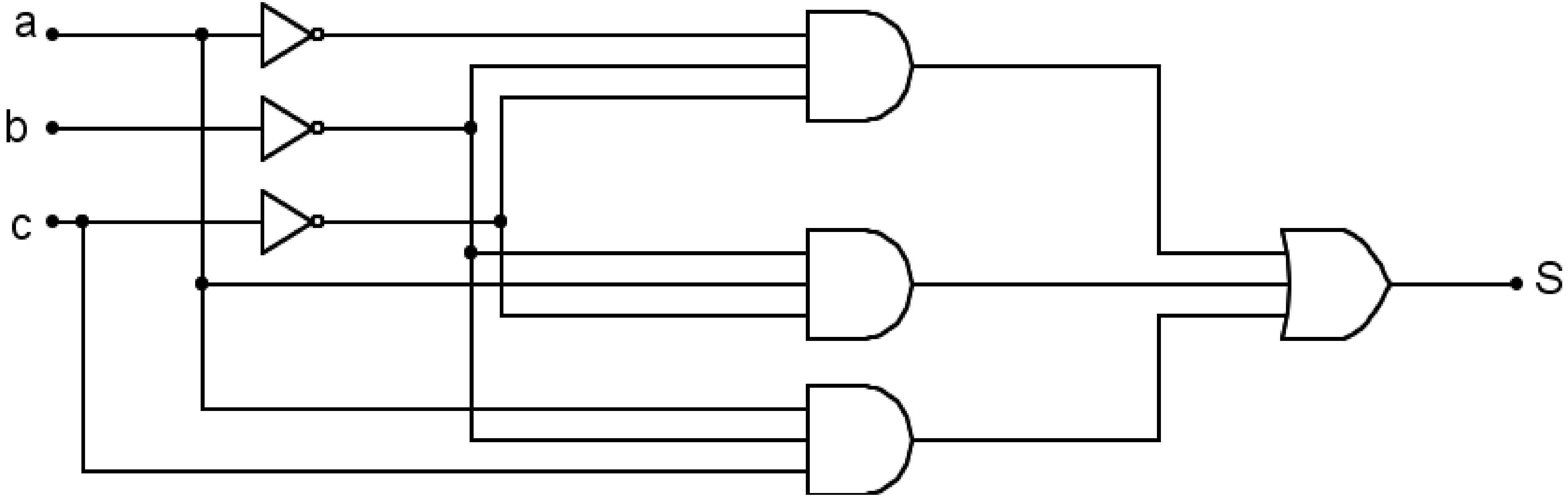
| | $a$ | $b$ | $c$ | $S$ | |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 | $m_0 = \bar{a}.\bar{b}.\bar{c}$ |
| 1 | 0 | 0 | 1 | 0 | $m_1 = \bar{a}.\bar{b}.c$ |
| 2 | 0 | 1 | 0 | 0 | $m_2 = \bar{a}.b.\bar{c}$ |
| 3 | 0 | 1 | 1 | 0 | $m_3 = \bar{a}.b.c$ |
| 4 | 1 | 0 | 0 | 1 | $m_4 = a.\bar{b}.\bar{c}$ |
| 5 | 1 | 0 | 1 | 1 | $m_5 = a.\bar{b}.c$ |
| 6 | 1 | 1 | 0 | 0 | $m_6 = a.b.\bar{c}$ |
| 7 | 1 | 1 | 1 | 0 | $m_7 = a.b.c$ |

$$\sum m = m_0 + m_4 + m_5 = \bar{a}.\bar{b}.\bar{c} + a.\bar{b}.\bar{c} + a.\bar{b}.c$$

# Lógica Digital

## Circuito resultante

$$\sum m = m_0 + m_4 + m_5 = \bar{a}.\bar{b}.\bar{c} + a.\bar{b}.\bar{c} + a.\bar{b}.c$$



# Lógica Digital

## Maxtermos

Podemos escrever o maxtermo de uma linha negando as variáveis que tem valor 1, mantendo as variáveis que tem valor 0, e realizando uma disjunção lógica entre elas. Assim, temos:

| | $a$ | $b$ | $c$ | $S$ |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 2 | 0 | 1 | 0 | 0 |
| 3 | 0 | 1 | 1 | 0 |
| 4 | 1 | 0 | 0 | 1 |
| 5 | 1 | 0 | 1 | 1 |
| 6 | 1 | 1 | 0 | 0 |
| 7 | 1 | 1 | 1 | 0 |

$$M_0 = a + b + c$$

$$M_1 = a + b + \bar{c}$$

$$M_2 = a + \bar{b} + c$$

$$M_3 = a + \bar{b} + \bar{c}$$

$$M_4 = \overline{a_1} + b + c$$

$$M_5 = \bar{a} + b + \bar{c}$$

$$M_6 = \bar{a} + \bar{b} + c$$

$$M_7 = \bar{a} + \bar{b} + \bar{c}$$

# Lógica Digital

## Produto de maxtermos

Para obter essa forma canônica, basta multiplicar as linhas cuja saída da tabela verdade é 0.

| | $a$ | $b$ | $c$ | $S$ |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 2 | 0 | 1 | 0 | 0 |
| 3 | 0 | 1 | 1 | 0 |
| 4 | 1 | 0 | 0 | 1 |
| 5 | 1 | 0 | 1 | 1 |
| 6 | 1 | 1 | 0 | 0 |
| 7 | 1 | 1 | 1 | 0 |

$M_0 = a + b + c$

$M_1 = a + b + \bar{c}$

$M_2 = a + \bar{b} + c$

$M_3 = a + \bar{b} + \bar{c}$

$M_4 = \overline{a_1} + b + c$

$M_5 = \bar{a} + b + \bar{c}$

$M_6 = \bar{a} + \bar{b} + c$

$M_7 = \bar{a} + \bar{b} + \bar{c}$

$$\sum M = M_1 . M_2 . M_3 . M_6 . M_7 =$$

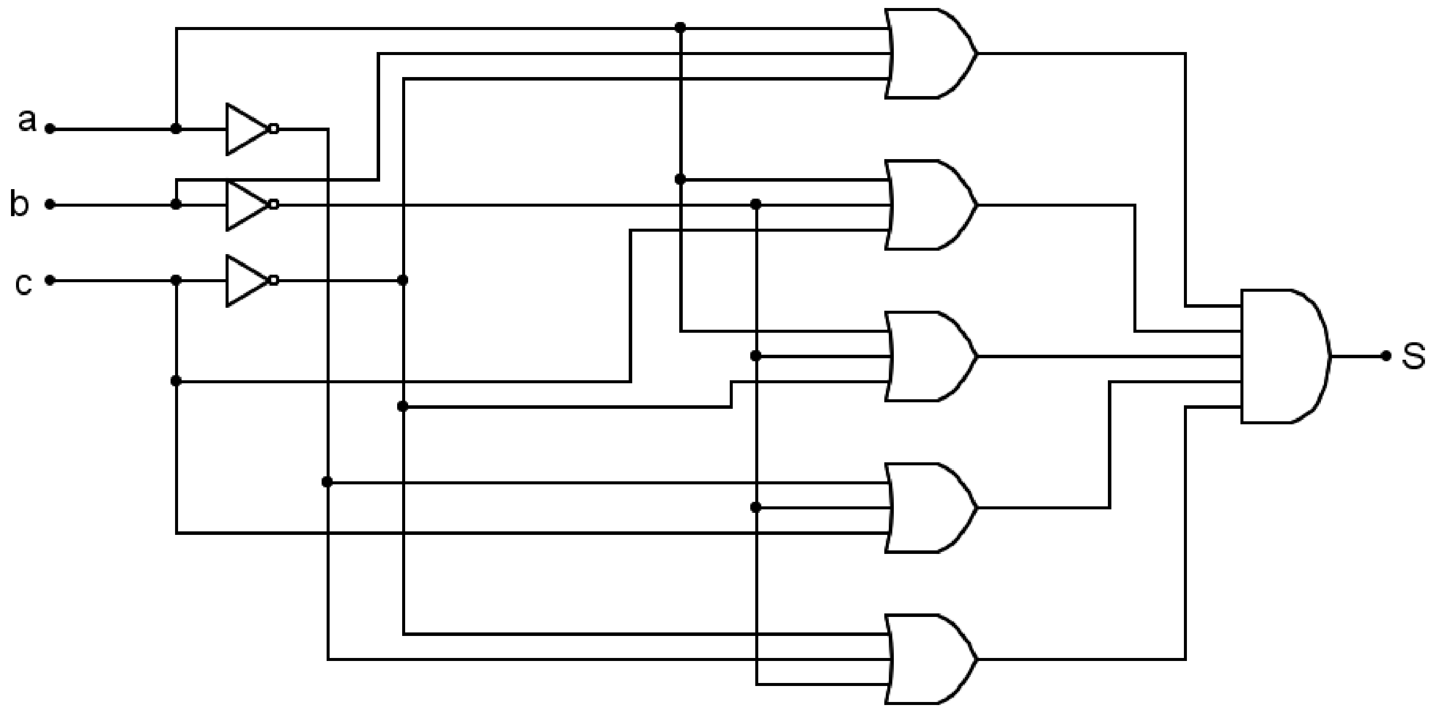
$$= (a + b + \bar{c}) . \ (a + \bar{b} + c) . (a + \bar{b} + \bar{c}) . (\bar{a} + \bar{b} + c) . (\bar{a} + \bar{b} + \bar{c})$$

# Lógica Digital

## Circuito resultante

$$\sum M = M_1 . M_2 . M_3 . M_6 . M_7 =$$

$$= (x_1 + x_2 + \overline{x_3}).(x_1 + \overline{x_2} + x_3).(x_1 + \overline{x_2} + \overline{x_3}).(\overline{x_1} + \overline{x_2} + x_3).(\overline{x_1} + \overline{x_2} + \overline{x_3})$$



# Aula 13:

# Mapas de Karnaugh

## Motivação

Mapa de Karnaugh são uma forma de se obter a expressão mínima a partir de uma tabela-verdade. Em relação à manipulação algébrica, mapas de Karnaugh são considerados superiores pela simplicidade e pela garantia de uma expressão mínima para situações de até 4 variáveis de entrada. Porém, para expressões com mais de 4 variáveis, esse método torna-se demasiadamente complexo.

A partir de um mapa de Karnaugh, podem ser obtidas duas expressões mínimas: a expressão na forma de produto das somas, e a expressão na forma de soma de produtos. Verificaremos como cada uma das expressões podem ser obtidas em diversas situações.

# Lógica Digital

## Mapas de 2 variáveis

Primeiro, numeramos as linhas da tabela-verdade.

| $n$ | $a$ | $b$ | $S$ |
|---|---|---|---|
| 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 2 | 1 | 0 | 1 |
| 3 | 1 | 1 | 1 |

# Lógica Digital

## Mapas de 2 variáveis

Primeiro, numeramos as linhas da tabela-verdade.

| $n$ | $a$ | $b$ | $S$ |
|---|---|---|---|
| 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 2 | 1 | 0 | 1 |
| 3 | 1 | 1 | 1 |

Depois, colocamos esses números em um mapa com essa forma, e preenchemos o mapa com os valores de saída.

| $a \backslash b$ | **0** | **1** |
|---|---|---|
| **0** | $_0$ | $_1$ |
| **1** | $_2$ | $_3$ |

| $a \backslash b$ | **0** | **1** |
|---|---|---|
| **0** | $1_0$ | $0_1$ |
| **1** | $1_2$ | $1_3$ |

# Lógica Digital

## Mapas de 2 variáveis

Em seguida, devemos circular, na tabela, grupos de números iguais que possuam uma quantidade de números equivalente a uma potência de 2 (1, 2, 4, 8 ou 16). Podemos optar por circular os uns ou os zeros. Assim, para as duas possibilidades, teremos:

| $a \backslash b$ | **0** | **1** |
|---|---|---|
| **0** | $1_0$ | $0_1$ |
| **1** | $1_2$ | $1_3$ |

| $a \backslash b$ | **0** | **1** |
|---|---|---|
| **0** | $1_0$ | $0_1$ |
| **1** | $1_2$ | $1_3$ |

# Lógica Digital

## Mapas de 2 variáveis

| $a \backslash b$ | 0 | 1 |
|---|---|---|
| 0 | $1_0$ | $0_1$ |
| 1 | $1_2$ | $1_3$ |

Soma de produtos (SOP)

Deve-se multiplicar as variáveis presentes dentro do grupo e somar os grupos.

$$a + \bar{b}$$

| $a \backslash b$ | 0 | 1 |
|---|---|---|
| 0 | $1_0$ | $0_1$ |
| 1 | $1_2$ | $1_3$ |

Produto das somas (POS)

Deve-se também multiplicar as variáveis presentes dentro do grupo, somar os grupos, e negar, por meio das leis de De Morgan, a expressão resultante

$$\overline{(\bar{a}.b)} = a + \bar{b}$$

Em ambas as situações, um grupo que contém $x = 1$ e $x = 0$ suprime a variável $x$.

# Lógica Digital

## Mapas de 3 variáveis

Primeiro, numeramos as linhas da tabela-verdade.

| $n$ | $a$ | $b$ | $c$ | $S$ |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 |
| 2 | 0 | 1 | 0 | 0 |
| 3 | 0 | 1 | 1 | 1 |
| 4 | 1 | 0 | 0 | 1 |
| 5 | 1 | 0 | 1 | 1 |
| 6 | 1 | 1 | 0 | 1 |
| 7 | 1 | 1 | 1 | 0 |

## Mapas de 3 variáveis

O mapa de Karnaugh resultante corresponde a uma tabela 2x4, que contém todas as possibilidades que as variáveis podem obter. Porém, para permitir que o mapa seja circulado, é necessário dispor as variáveis da seguinte forma:

| $c \backslash ab$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| **0** | 0 | 2 | 6 | 4 |
| **1** | 1 | 3 | 7 | 5 |

Depois, preenchemos os valores:

| $c \backslash ab$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| **0** | $0_{0}$ | $0_{2}$ | $1_{6}$ | $1_{4}$ |
| **1** | $1_{1}$ | $1_{3}$ | $0_{7}$ | $1_{5}$ |

# Lógica Digital

## Mapas de 3 variáveis

Soma de produtos (SOP)

| $c \backslash ab$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| **0** | $0_{0}$ | $0_{2}$ | $1_{6}$ | $1_{4}$ |
| **1** | $1_{1}$ | $1_{3}$ | $0_{7}$ | $1_{5}$ |

Deve-se multiplicar as variáveis presentes dentro do grupo e somar os grupos.

$$\bar{a}c + a\bar{c} + a\bar{b}$$

Produto das somas (POS)

| $c \backslash ab$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| **0** | $0_{0}$ | $0_{2}$ | $1_{6}$ | $1_{4}$ |
| **1** | $1_{1}$ | $1_{3}$ | $0_{7}$ | $1_{5}$ |

Deve-se também multiplicar as variáveis presentes dentro do grupo, somar os grupos, e negar, por meio das leis de De Morgan, a expressão resultante

$$\overline{(\bar{a}.\bar{c}) + (a.b.c.)} = (a + c).(\bar{a} + \bar{b} + \bar{c})$$

Em ambas as situações, um grupo que contém $x = 1$ e $x = 0$ suprime a variável $x$.

# Lógica Digital

## Mapas de 3 variáveis

Primeiro, numeramos as linhas da tabela-verdade.

| $n$ | $a$ | $b$ | $c$ | $d$ | $S$ |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 |
| 2 | 0 | 0 | 1 | 0 | 0 |
| 3 | 0 | 0 | 1 | 1 | 0 |
| 4 | 0 | 1 | 0 | 0 | 0 |
| 5 | 0 | 1 | 0 | 1 | 0 |
| 6 | 0 | 1 | 1 | 0 | 1 |
| 7 | 0 | 1 | 1 | 1 | 0 |
| 8 | 1 | 0 | 0 | 0 | 1 |
| 9 | 1 | 0 | 0 | 1 | 1 |
| 10 | 1 | 0 | 1 | 0 | 1 |
| 11 | 1 | 0 | 1 | 1 | 1 |
| 12 | 1 | 1 | 0 | 0 | 1 |
| 13 | 1 | 1 | 0 | 1 | 1 |
| 14 | 1 | 1 | 1 | 0 | 1 |
| 15 | 1 | 1 | 1 | 1 | 0 |

# Lógica Digital

## Mapas de 3 variáveis

Depois, numeramos e preenchemos o mapa.

| $n$ | $a$ | $b$ | $c$ | $d$ | $S$ |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 |
| 2 | 0 | 0 | 1 | 0 | 0 |
| 3 | 0 | 0 | 1 | 1 | 0 |
| 4 | 0 | 1 | 0 | 0 | 0 |
| 5 | 0 | 1 | 0 | 1 | 0 |
| 6 | 0 | 1 | 1 | 0 | 1 |
| 7 | 0 | 1 | 1 | 1 | 0 |
| 8 | 1 | 0 | 0 | 0 | 1 |
| 9 | 1 | 0 | 0 | 1 | 1 |
| 10 | 1 | 0 | 1 | 0 | 1 |
| 11 | 1 | 0 | 1 | 1 | 1 |
| 12 | 1 | 1 | 0 | 0 | 1 |
| 13 | 1 | 1 | 0 | 1 | 1 |
| 14 | 1 | 1 | 1 | 0 | 1 |
| 15 | 1 | 1 | 1 | 1 | 0 |

| $ab \backslash cd$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0 | 1 | 3 | 2 |
| 01 | 4 | 5 | 7 | 6 |
| 11 | 12 | 13 | 15 | 14 |
| 10 | 8 | 9 | 11 | 10 |

| $ab \backslash cd$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | $0_{0}$ | $0_{1}$ | $0_{3}$ | $0_{2}$ |
| 01 | $0_{4}$ | $0_{5}$ | $0_{7}$ | $1_{6}$ |
| 11 | $1_{12}$ | $1_{13}$ | $1_{15}$ | $1_{14}$ |
| 10 | $1_{8}$ | $1_{9}$ | $1_{11}$ | $1_{10}$ |

# Lógica Digital

## Mapas de 3 variáveis

Soma de produtos (SOP)

| $ab \backslash cd$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | $0_{0}$ | $0_{1}$ | $0_{3}$ | $0_{2}$ |
| 01 | $0_{4}$ | $0_{5}$ | $0_{7}$ | $1_{6}$ |
| 11 | $1_{12}$ | $1_{13}$ | $1_{15}$ | $1_{14}$ |
| 10 | $1_{8}$ | $1_{9}$ | $1_{11}$ | $1_{10}$ |

Deve-se multiplicar as variáveis presentes dentro do grupo e somar os grupos.

$$a + (b.c.\bar{d})$$

Produto das somas (POS)

| $ab \backslash cd$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | $0_{0}$ | $0_{1}$ | $0_{3}$ | $0_{2}$ |
| 01 | $0_{4}$ | $0_{5}$ | $0_{7}$ | $1_{6}$ |
| 11 | $1_{12}$ | $1_{13}$ | $1_{15}$ | $1_{14}$ |
| 10 | $1_{8}$ | $1_{9}$ | $1_{11}$ | $1_{10}$ |

Deve-se também multiplicar as variáveis presentes dentro do grupo, somar os grupos, e negar, por meio das leis de De Morgan, a expressão resultante

$$\overline{\bar{a}.\bar{b} + \bar{a}\bar{c} + \bar{a}d} = (a + b).(a + b).(a + \bar{d}).$$

Em ambas as situações, um grupo que contém $x = 1$ e $x = 0$ suprime a variável $x$.

# Lógica Digital

## Diretrizes gerais para marcar grupos

Já sabemos o que deve ser feito, mas, talvez, você ainda tenha algumas dúvidas a respeito de como circular conjuntos. Portanto, seguem algumas diretrizes:

1. As regiões circuladas sempre precisam conter um número de uns ou zeros potência de dois, como 1, 2, 4, 8 e 16.

| $ab \backslash cd$ | **00** | **01** | **11** | **10** |
|---|---|---|---|---|
| **00** | $0_{0}$ | $0_{1}$ | $0_{3}$ | $0_{2}$ |
| **01** | $0_{4}$ | $0_{5}$ | $0_{7}$ | $1_{6}$ |
| **11** | $1_{12}$ | $1_{13}$ | $1_{15}$ | $1_{14}$ |
| **10** | $1_{8}$ | $1_{9}$ | $1_{11}$ | $1_{10}$ |

**NÃO PODE!!!**

# Lógica Digital

## Diretrizes gerais para marcar grupos

Já sabemos o que deve ser feito, mas, talvez, você ainda tenha algumas dúvidas a respeito de como circular conjuntos. Portanto, seguem algumas diretrizes:

2. As únicas formas permitidas (considerando-se quando os grupos extrapolam as margens da tabela) são os retângulos e quadrados.

| ab \ cd | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | $0_{0}$ | $0_{1}$ | $0_{3}$ | $0_{2}$ |
| 01 | $0_{4}$ | $0_{5}$ | $0_{7}$ | $1_{6}$ |
| 11 | $1_{12}$ | $1_{13}$ | $1_{15}$ | $1_{14}$ |
| 10 | $1_{8}$ | $1_{9}$ | $1_{11}$ | $1_{10}$ |

**NÃO PODE!!!**

# Lógica Digital

## Diretrizes gerais para marcar grupos

Já sabemos o que deve ser feito, mas, talvez, você ainda tenha algumas dúvidas a respeito de como circular conjuntos. Portanto, seguem algumas diretrizes:

3.Os grupos devem ter o maior tamanho possível.

| $ab \backslash cd$ | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| **00** | $0_{0}$ | $0_{1}$ | $0_{3}$ | $0_{2}$ |
| **01** | $0_{4}$ | $0_{5}$ | $0_{7}$ | $1_{6}$ |
| **11** | $1_{12}$ | $1_{13}$ | $1_{15}$ | $1_{14}$ |
| **10** | $1_{8}$ | $1_{9}$ | $1_{11}$ | $1_{10}$ |

**NÃO PODE!!!**

## Diretrizes gerais para marcar grupos

Já sabemos o que deve ser feito, mas, talvez, você ainda tenha algumas dúvidas a respeito de como circular conjuntos. Portanto, seguem algumas diretrizes:

4. Um grupo obrigatoriamente precisa marcar alguma variável que não foi marcada antes.

| ab \ cd | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | $0_0$ | $0_1$ | $0_3$ | $0_2$ |
| 01 | $0_4$ | $0_5$ | $0_7$ | $1_6$ |
| 11 | $1_{12}$ | $1_{13}$ | $1_{15}$ | $1_{14}$ |
| 10 | $1_8$ | $1_9$ | $1_{11}$ | $1_{10}$ |

**NÃO PODE!!!**

# Lógica Digital

## Diretrizes gerais para marcar grupos

Já sabemos o que deve ser feito, mas, talvez, você ainda tenha algumas dúvidas a respeito de como circular conjuntos. Portanto, seguem algumas diretrizes:

1. As regiões circuladas sempre precisam conter um número de uns ou zeros potência de dois, como 1, 2, 4, 8 e 16.

2. As únicas formas permitidas (considerando-se quando os grupos extrapolam as margens da tabela) são os retângulos e quadrados.

3. Os grupos devem ter o maior tamanho possível.

4. A implementação mínima é aquela que usa o menor número de grupos.

A implementação mínima é aquela que marca o menor número de grupos. Seguindo essas regras, você provavelmente obterá uma implementação mínima.

# Lógica Digital

## Motivação

***"Dois sensores informam a temperatura de uma sala. Um deles, $a$ retorna 1 sempre que a temperatura for maior que 16°C, e o outro, $b$ retorna 1 sempre que a temperatura for maior que 18°C. Construa um circuito que devolva 1 sempre que a temperatura estiver entre 16°C e 18°C."***

| $n$ | $a$ | $b$ | $S$ |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | ? |
| 2 | 1 | 0 | 1 |
| 3 | 1 | 1 | 0 |

A situação na linha 1 nunca vai acontecer! A temperatura nunca pode ser menor que 16°C e maior que 18°C ao mesmo tempo!

Assim, a saída do circuito nessa condição simplesmente não importa. A chamamos de *don't care*.

| $n$ | $s_{16}$ | $s_{18}$ | $S$ |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | d |
| 2 | 1 | 0 | 1 |
| 3 | 1 | 1 | 0 |

# Lógica Digital

## Motivação

Um don’t care pode ter o valor que acharmos mais conveniente para nosso mapa.

| a \ b | 0 | 1 |
|---|---|---|
| **0** | $0_0$ | $d_1$ |
| **1** | $1_2$ | $0_3$ |

| a \ b | 0 | 1 |
|---|---|---|
| **0** | $0_0$ | $d_1$ |
| **1** | $1_2$ | $0_3$ |

$$\overline{(\bar{A} + B)} = A.\bar{B}$$

## Motivação

Veja outro exemplo, com 4 variáveis.

| ab \ cd | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| **00** | $1_{0}$ | $1_{1}$ | $1_{3}$ | $0_{2}$ |
| **01** | $1_{4}$ | $d_{5}$ | $d_{7}$ | $0_{6}$ |
| **11** | $0_{12}$ | $0_{13}$ | $0_{15}$ | $0_{14}$ |
| **10** | $0_{8}$ | $0_{9}$ | $0_{11}$ | $0_{10}$ |

| ab \ cd | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| **00** | $1_{0}$ | $1_{1}$ | $1_{3}$ | $0_{2}$ |
| **01** | $1_{4}$ | $d_{5}$ | $d_{7}$ | $0_{6}$ |
| **11** | $0_{12}$ | $0_{13}$ | $0_{15}$ | $0_{14}$ |
| **10** | $0_{8}$ | $0_{9}$ | $0_{11}$ | $0_{10}$ |

# Aula 14:

# Representação binária

# Lógica Digital

## Recordando o sistema decimal

O sistema decimal é composto por dez algarismos, 0, 1, 2, 3, 4, 5, 6, 7, 8 e 9, cada um deles representando uma dada quantidade de itens. A base de nosso sistema é, portanto, o número 10. Assim, temos:

$$1708 = 1 \times 10^3 + 7 \times 10^2 + 0 \times 10^1 + 8 \times 10^0 = 1000 + 700 + 0 + 8 = 1708$$

# Lógica Digital

## Recordando o sistema decimal

O sistema decimal é composto por dez algarismos, 0, 1, 2, 3, 4, 5, 6, 7, 8 e 9, cada um deles representando uma dada quantidade de itens. A base de nosso sistema é, portanto, o número 10. Assim, temos:

$$1708 = 1 \times 10^3 + 7 \times 10^2 + 0 \times 10^1 + 8 \times 10^0 = 1000 + 700 + 0 + 8 = 1708$$

## Aplicando no sistema binário

O mesmo princípio vale para o sistema binário: o sistema binário é composto por dois algarismos, 0 e 1 e sua base é, portanto, o número 2, ou 10, em binário. Isso significa que temos:

$$11101 = 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 16 + 8 + 4 + 0 + 1 = 29$$

# Lógica Digital

## Escrevendo em binário

Podemos escrever um número em binário através de um algoritmo prático: deve-se dividir o número por 2 sucessivamente, e anotar o resto de cada divisão. O número será formado pela aglutinação de todos os restos, com o último resto obtido mais à esquerda, e o primeiro mais à direita. Assim, temos o dispositivo prático:

| | | | | Quociente | | Resto |
|---|---|---|---|---|---|---|
| 29 | ÷ | 2 | = | 14 | + | 1 |
| 14 | ÷ | 2 | = | 7 | + | 0 |
| 7 | ÷ | 2 | = | 3 | + | 1 |
| 3 | ÷ | 2 | = | 1 | + | 1 |
| 1 | ÷ | 2 | = | 0 | + | 1 |
| 0 | ÷ | 2 | = | 0 | + | 0 |

↑ Mais à direita

Mais à esquerda

$$11101$$

# Lógica Digital

## Escrevendo em binário

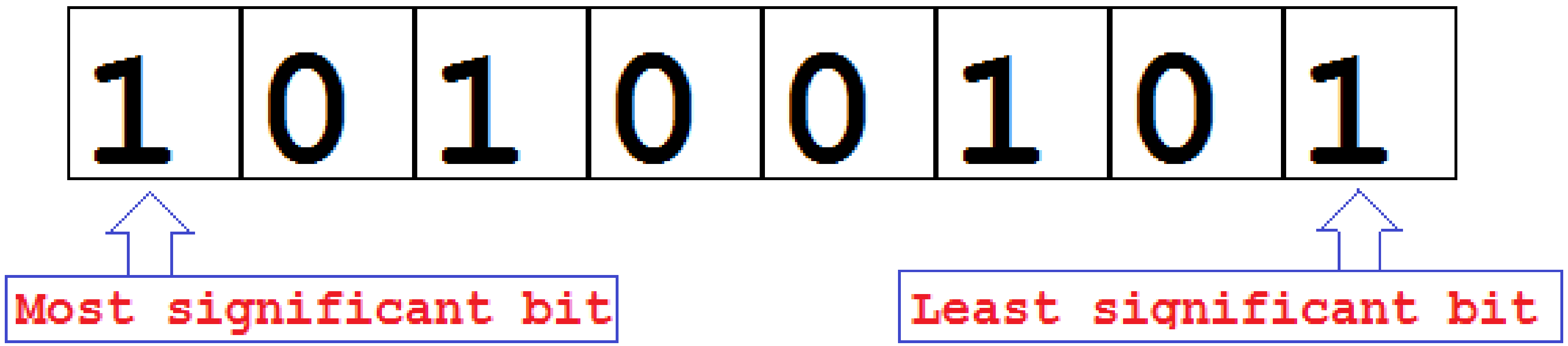
Para fazer o caminho inverso, também temos um dispositivo prático: deve-se escrever o número binário na vertical, e multiplicar o enésimo algarismo do número binário (de cima para baixo) por $2^n$ e somar os resultados.

$$
\begin{array}{lllr}
1 & \times 2\text{^} & 0 = & 1 \\
0 & \times 2\text{^} & 1 = & 0 \\
1 & \times 2\text{^} & 2 = & 4 \\
1 & \times 2\text{^} & 3 = & 8 \\
1 & \times 2\text{^} & 4 = & 16 \\
0 & \times 2\text{^} & 5 = & \underline{\;\;0} \\
 & & & 29
\end{array}
$$

# Lógica Digital

## MSB e LSB

Em um número binário, chamamos o bit mais a esquerda de bit mais significativo, ou MSB (*most significant bit*), e chamamos o bit mais a direita de bit menos significativo, ou LSB (*less significant bit)*. Chamamos um conjunto de 8 bits de um byte.



# Lógica Digital

## Número de binários que podemos armazenar

Dada uma quantidade de $b$ bits, o número $n$ de números binários que podemos representar utilizando esses bits é dado pela fórmula $n = 2^b$ e corresponde aos números entre 0 e $2^b - 1$ . Assim, com 4 bits, podemos representar 16 números, entre 0 e 15. São eles:

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

$$2 \times 2 \times 2 \times 2 = 16$$

# Lógica Digital

## Adição de números binários

Recordando a adição em decimal:

$$
\begin{array}{r}
11\phantom{0} \\
27 \\
+\quad 85 \\
\hline
112
\end{array}
$$

# Lógica Digital

## Adição de números binários

Podemos usar o mesmo algoritmo em binário:

$$
\begin{array}{cccccccc}
 & 1 & 1 & 1 & & & & \\
 & & & 1 & 1 & 1 & 0 & 1 \\
+ & & 1 & 1 & 1 & 0 & 1 & 0 \\
\hline
 & 1 & 0 & 1 & 0 & 1 & 1 & 1
\end{array}
$$

# Lógica Digital

## Complemento de 2

No sistema decimal, utilizamos o sinal de menos na frente de um número para indicar que este é negativo. Porém, em sistemas digitais, não existem artifícios para inserir este sinal na frente de um número binário negativo.

Ainda assim, evidentemente precisamos trabalhar com binários negativos. Também é interessante que a operação de adição possa funcionar como uma operação de subtração quando um número negativo é adicionado em um dos fatores a serem somados.

A forma escolhida para tal representação é a de complemento de 2.

## Complemento de 2

Ao escrever um número de $n$ bits nessa forma, iremos utilizar o bit mais significativo como um indicador de sinal. Isso significa que, a partir de agora, números positivos possuirão um zero à esquerda em toda situação. Assim, o número 4 precisa, agora, de 4 bits (ao invés de 3) para ser representado, da forma $0100$.

Um número negativo pode ser obtido na forma de complemento de dois por um algoritmo prático: deve-se inverter todos os bits (trocando 1 por 0 e 0 por 1) e, em seguida, somar 1 ao resultado invertido. Assim, temos:

$$\begin{array}{l} 0100 \\ \downarrow \ \text{Inverter} \\ 1011 \\ \downarrow \ \text{Somar 1} \\ 1100 \end{array}$$

# Lógica Digital

## Complemento de 2

O processo para converter um número em complemento de 2 para binário sem sinal (de forma a descobrir qual número corresponde) é exatamente o mesmo: primeiro invertê-lo, depois somar 1. Veja:

$$
\begin{array}{c}
1100 \\
\downarrow \text{ Inverter} \\
0011 \\
\downarrow \text{ Somar 1} \\
0100
\end{array}
$$

## Subtração em binário

A grande vantagem de utilizarmos essa forma de representação é que, agora, podemos subtrair números binários da mesma forma que os somamos, utilizando exatamente o mesmo processo. Veja, por exemplo, a operação $6 - 4 = 2$:

$$
\begin{array}{r}
11\phantom{000} \\
0110 \\
+\;1100 \\
\hline
10010
\end{array}
$$

# Lógica Digital

## Subtração em binário

Aqui, temos um detalhe importante: esperávamos um resultado com 4 bits (três para o número, um para o sinal), mas obtivemos um resultado composto por 5 bits. Esse último bit é denominado *ignore* e, como o nome diz, deve ser ignorado sempre que o resultado retorne um número para além do bit de sinal.

$$
\begin{array}{r}
11\phantom{00} \\
0110 \\
+\ 1100 \\
\hline
10010
\end{array}
$$

Ignore

# Lógica Digital

## Overflow aritmético

Algumas vezes, somamos dois números cujo resultado excede a quantidade de bits que podemos usar. Suponha, por exemplo, que queiramos somar 4 e 6 de forma a obter um resultado que caiba em 4 bits com sinal. Teríamos a seguinte operação:

$$\begin{array}{r} 0100 \\ +\ 0110 \\ \hline 1010 \end{array}$$

# Lógica Digital

## Overflow aritmético

Algumas vezes, somamos dois números cujo resultado excede a quantidade de bits que podemos usar. Suponha, por exemplo, que queiramos somar 4 e 6 de forma a obter um resultado que caiba em 4 bits com sinal. Teríamos a seguinte operação:

$$\begin{array}{r} 0100 \\ +\ 0110 \\ \hline 1010 \end{array}$$

1010, em binário sem sinal, é 10.

Porém, em complemento de 2 é -6.

Isso acontece porque possuímos apenas 3 bits para o número e 1 para o sinal, e o número 10 requer 4 bits para o número.

Portanto, chamamos essa situação de overflow aritmético.

## Overflow aritmético

Algumas vezes, somamos dois números cujo resultado excede a quantidade de bits que podemos usar. Suponha, por exemplo, que queiramos somar 4 e 6 de forma a obter um resultado que caiba em 4 bits com sinal. Teríamos a seguinte operação:

$$\begin{array}{r} 0100 \\ +\ 0110 \\ \hline 1010 \end{array}$$

1010, em binário sem sinal, é 10.

Porém, em complemento de 2 é -6.

Isso acontece porque possuímos apenas 3 bits para o número e 1 para o sinal, e o número 10 requer 4 bits para o número.

Portanto, chamamos essa situação de overflow aritmético.

Para solucioná-lo, adicionamos mais 1 bit à operação.

$$\begin{array}{r} 00100 \\ +\ 00110 \\ \hline 01010 \end{array}$$

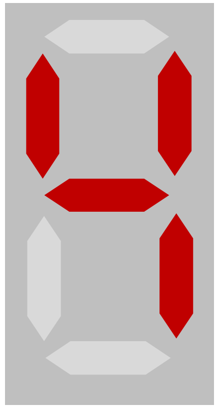
# Aula 15:

# Displays e Decoders

# Lógica Digital

## Motivação

Qual dos números é mais fácil de ler?

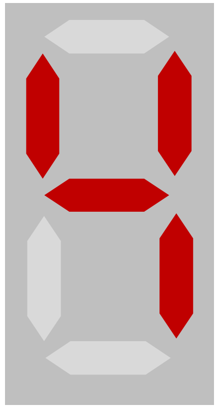


**ou**

# Lógica Digital

## Motivação

Qual dos números é mais fácil de ler?



**ou**

A forma binária é ótima para máquinas e péssima para humanos.

É interessante poder representar um número binário na forma decimal.

# Lógica Digital

## Motivação

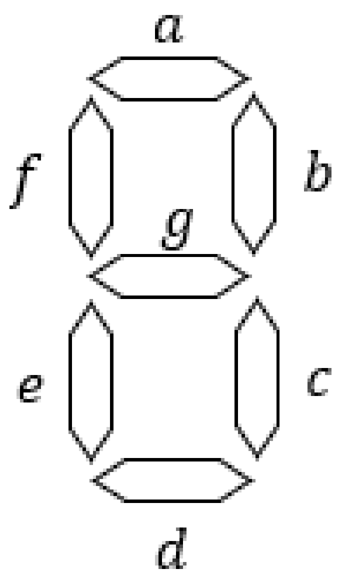
O uso de displays de 7 segmentos é uma das principais formas de fazê-lo.

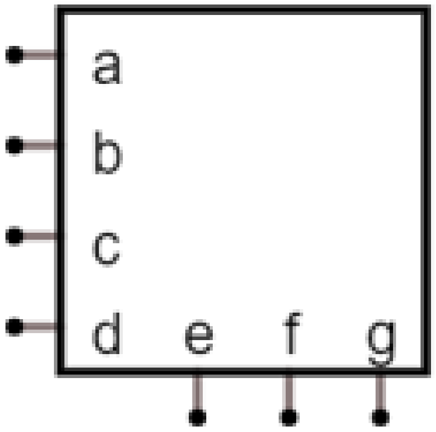
# Lógica Digital

## Motivação

Esse circuito tem 7 LEDs e 7 pinos, correspondentes a cada um dos LEDs.

Opcionalmente, pode também haver um LED para a vírgula ou ponto.



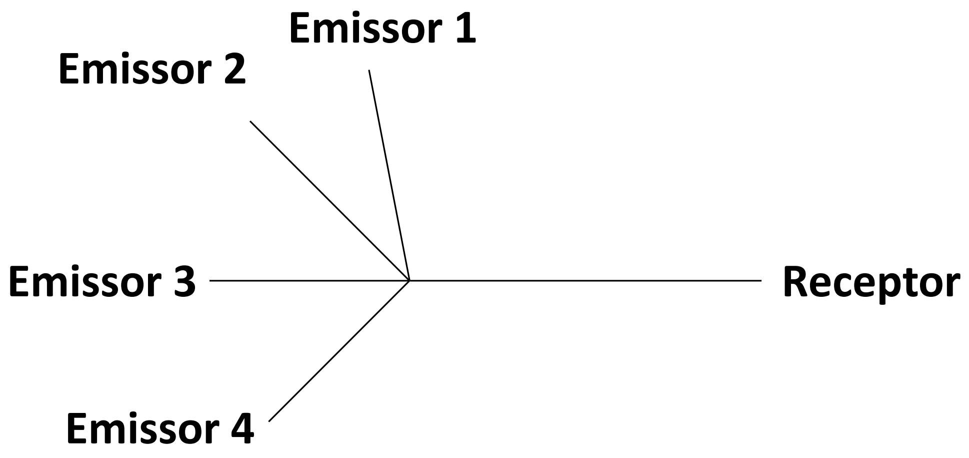
# Aula 16:

# Multiplexadores

# Lógica Digital

## Motivação

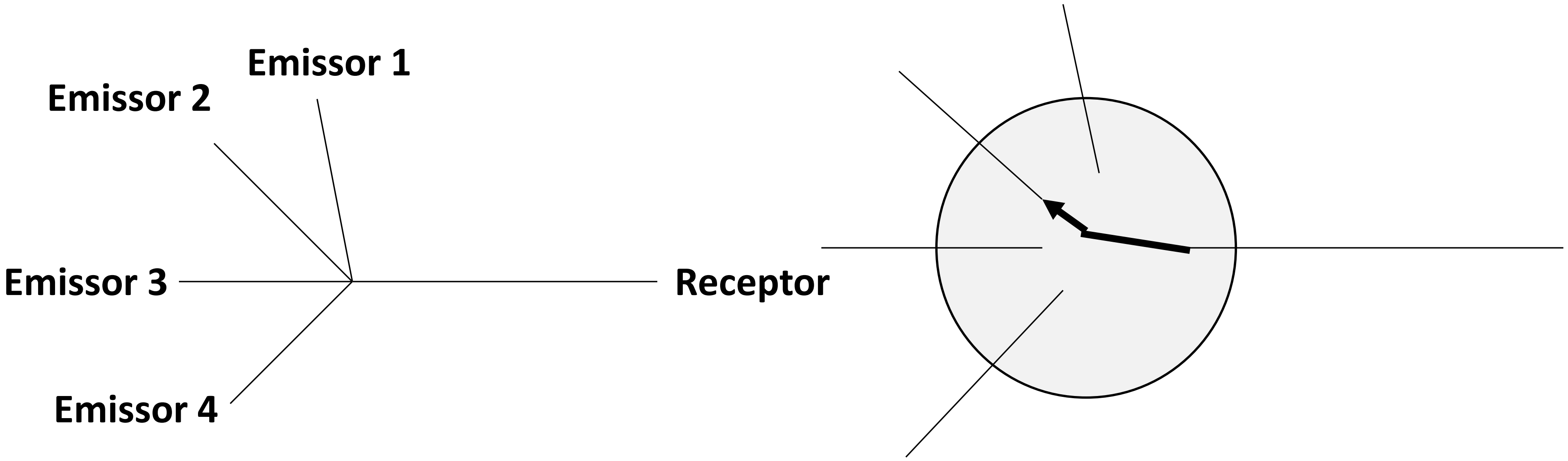
*Temos 4 pessoas que podem enviar código morse para um receptor em uma linha compartilhada. O receptor escolhe de quem ele deseja receber. Como configurar essa situação?*



# Lógica Digital

## Motivação

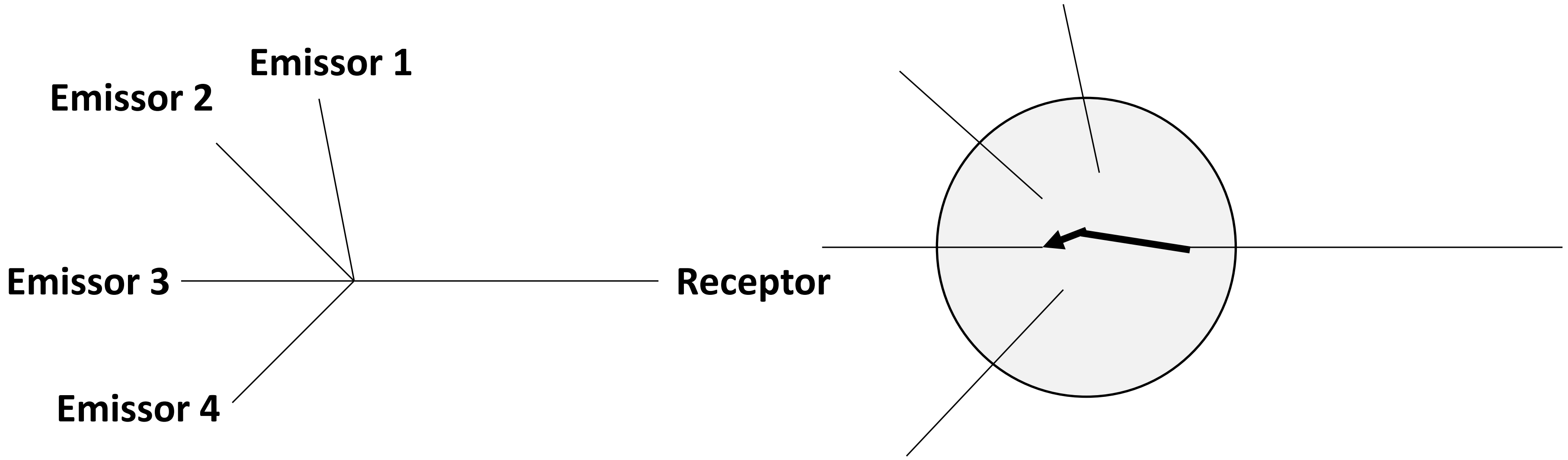
*Temos 4 pessoas que podem enviar código morse para um receptor em uma linha compartilhada. O receptor escolhe de quem ele deseja receber. Como configurar essa situação?*



# Lógica Digital

## Motivação

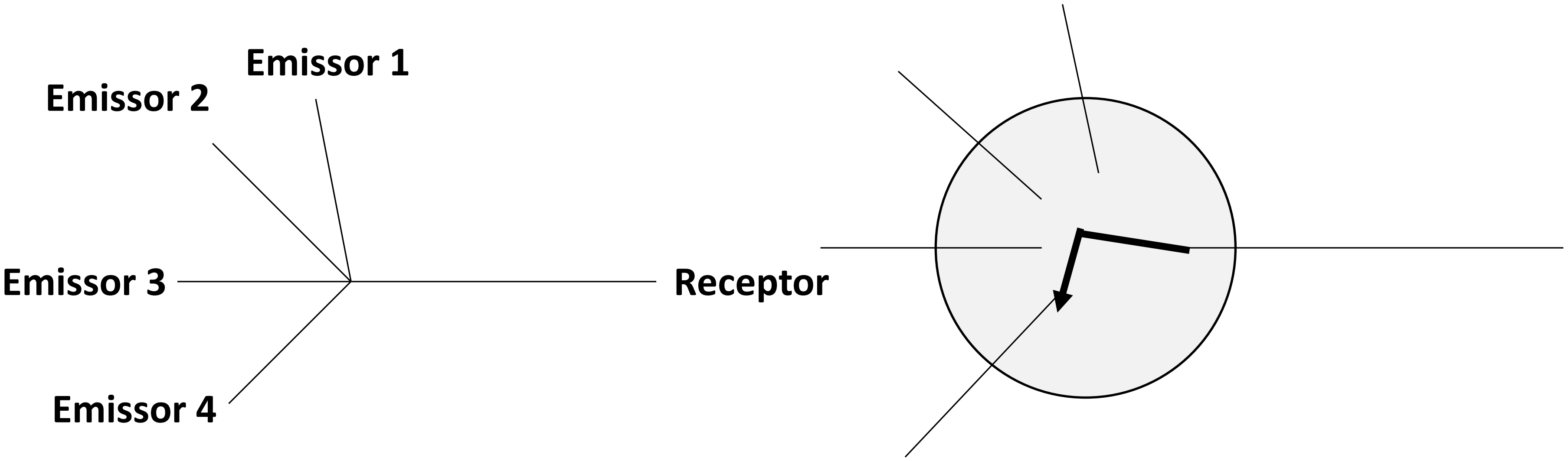
*Temos 4 pessoas que podem enviar código morse para um receptor em uma linha compartilhada. O receptor escolhe de quem ele deseja receber. Como configurar essa situação?*



# Lógica Digital

## Motivação

*Temos 4 pessoas que podem enviar código morse para um receptor em uma linha compartilhada. O receptor escolhe de quem ele deseja receber. Como configurar essa situação?*



# Aula 17:

# Somadores/Subtratores

# Aula 17:

# Somadores/Subtratores

# Lógica Digital

## Carry-in e Carry out

$$
\begin{array}{r}
11 \\
11 \\
\hline
110
\end{array}
$$

# Aula 18:

# Latch SR

# Lógica Digital

## Motivação

*“Um agricultor possui um galinheiro, dentro do qual existem galinhas que, as vezes, escapam. Há um sensor de presença no caminho que dá acesso ao galinheiro, e esse sensor devolve 1 no momento em que alguma galinha passa por ele, e devolve 0 logo em seguida. Projete um sistema que informe o agricultor, ao acordar, se alguma galinha escapou durante a noite”*

# Lógica Digital

## Motivação

Precisamos armazenar o input p = 1.

Como fazer isso?

# Lógica Digital

## Latch SR

Um circuito denominado latch set-reset (ou latch SR) é um circuito capaz de armazenar um bit de informação enquanto ele se mantiver ligado. Um latch SR pode ser sensível a um sinal alto (1) ou a um sinal baixo (0). Todo latch possui duas entradas, um set e um reset, e duas saídas, uma saída principal (Q) e uma saída negada (Q’).

Um latch SR sensível a sinal alto é denominado active high. Em seu estado inicial, ambas as entradas começam com sinal 0. Um sinal 1 no reset faz com que a entrada Q seja levada a 0, e a entrada Q’ seja levada a 1. Um sinal 1 no set faz com que a entrada Q seja levada a 1 e a entrada Q’ seja levada a 0. Por mais que o sinal no set seja levado para 0, as entradas continuarão definidas, até que um sinal de reset seja enviado.

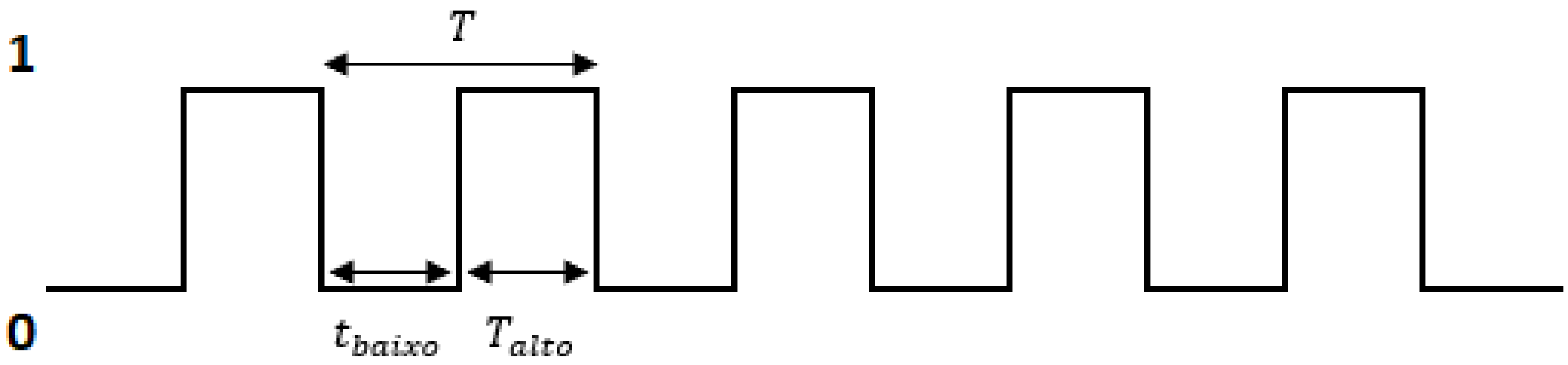
# Aula 19:

# Sinais de clock

# Lógica Digital

## Definição

Um clock é um sinal que oscila entre dois estados (1 e 0 no nosso caso) em um intervalo de tempo constante. Se fizermos um gráfico relacionando o tempo com o valor lido no clock, teríamos:



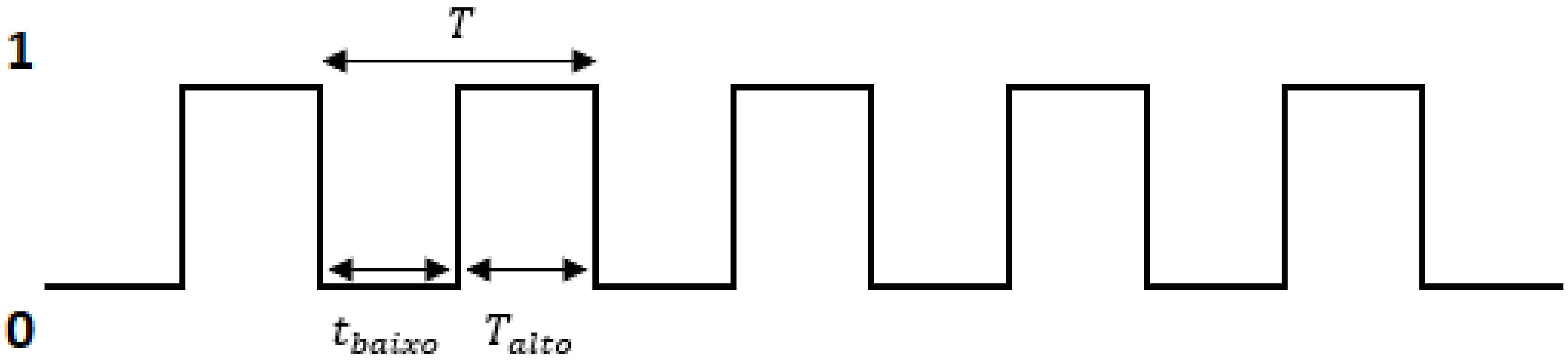
# Lógica Digital

## Definição

$$T = t_{baixo} + t_{alto}$$

$$\text{ciclo de trabalho} = \frac{t_{alto}}{T} \times 100$$

$$\text{frequência} = \frac{1}{T}$$



# Aula 20:

# Flip Flops RS, D, T e JK

# Aula 21:

# Registradores simples

# Aula 22:

# Registradores de deslocamento

# Aula 23:

# Contador decrescente e encerramento