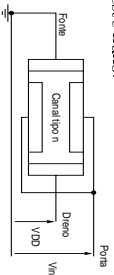


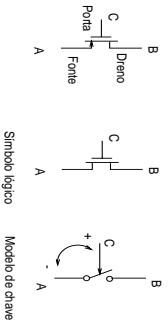
Complemento do Capítulo 3

Um transistor MOS é constituído por uma barra de material semicondutor (do tipo p ou do tipo n) e três terminais designados fonte, dreno e porta. A corrente flui ao longo da barra devido a uma fonte de tensão ligada entre fonte e dreno. O campo eléctrico criado pela tensão aplicada na porta pode aumentar ou barrar este fluxo.

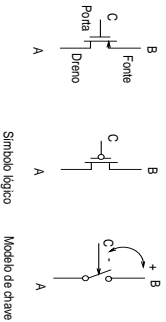


Os transistores (chaves eletrónicas) MOS podem ser considerados um dispositivo de três terminais que age como um potenciómetro controlado por tensão. Existem dois tipos de transistores MOS:

Tipo n (NMOS) : a chave está aberta (0) se $V_{CA} < V_{limiar}$ e fechada (1) se $V_{CA} > V_{limiar}$.



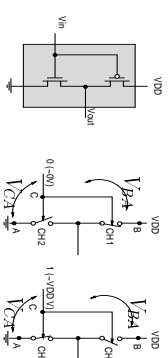
Tipo p (PMOS) : a chave está aberta (0) se $V_{BC} < V_{limiar}$ e fechada (1) se $V_{BC} > V_{limiar}$.



sendo V_{limiar} a tensão limiar para a condução da chave.

A realização das principais portas lógicas da família CMOS com uso de transistores nMOS e pMOS é apresentada a seguir.

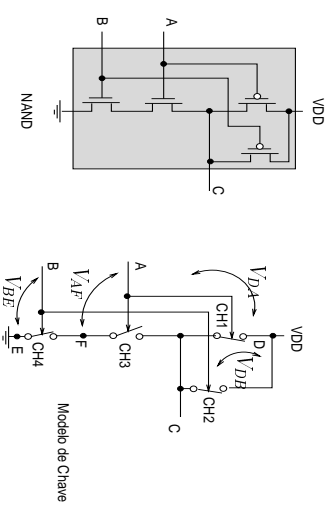
NOT :



| Vin | CH1 | CH2 | Vout |
|-----------|-----|-----|-----------|
| 0 V (0) | F | A | VDD V (1) |
| VDD V (1) | A | F | 0 V (0) |

F: fechada e A: aberta.

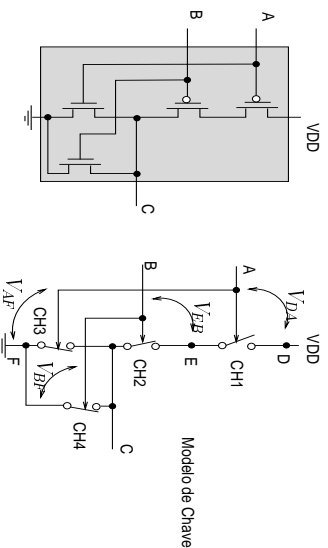
NAND :



Notam que quando B=0 (0 V) a chave CH4 fica aberta, a tensão V_{AF} é indefinida e, portanto, não se pode afirmar nada em relação ao estado da chave CH3.

| A | B | CH1 | CH2 | CH3 | CH4 | C |
|-----------|-----------|-----|-----|-----|-----|-----------|
| 0 V (0) | 0 V (0) | F | F | ? | A | VDD V (1) |
| 0 V (0) | VDD V (1) | F | A | A | F | VDD V (1) |
| VDD V (1) | 0 V (0) | A | F | ? | A | VDD V (1) |
| VDD V (1) | VDD V (1) | A | A | F | F | 0 V (0) |

NOR :



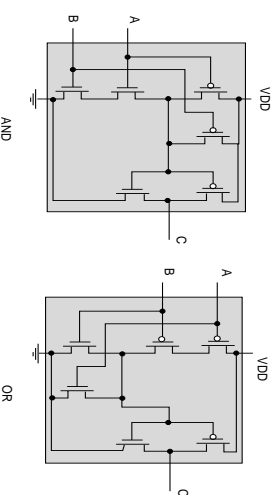
Notem que quando $A=1$ (V_{DD} V) a chave CH1 fica aberta, a tensão V_{nB} é indefinida e, portanto, não se pode afirmar nada em relação ao estado da chave CH2.

| A | B | CH1 | CH2 | CH3 | CH4 | C |
|----------------|----------------|-----|-----|-----|-----|----------------|
| 0 V (0) | 0 V (0) | F | F | A | A | V_{DD} V (1) |
| 0 V (0) | V_{DD} V (1) | F | A | A | F | 0 V (0) |
| V_{DD} V (1) | 0 V (0) | A | ? | F | A | 0 V (0) |
| V_{DD} V (1) | V_{DD} V (1) | A | ? | F | F | 0 V (0) |

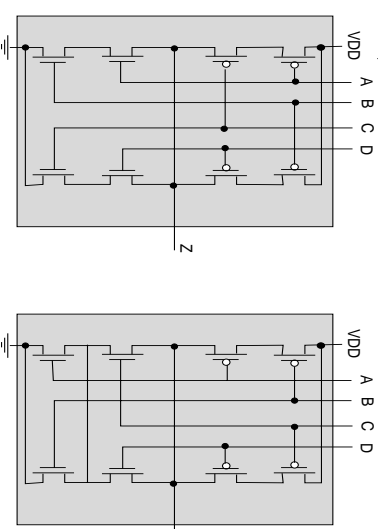
A combinação destes esquemas básicos nos permitem “construir” outras funções lógicas com uso de transistores, como as portas AND e OR. A partir das portas NAND e NOR podemos implementar portas AND e OR, sabendo que

$$ab = ((ab)')'$$

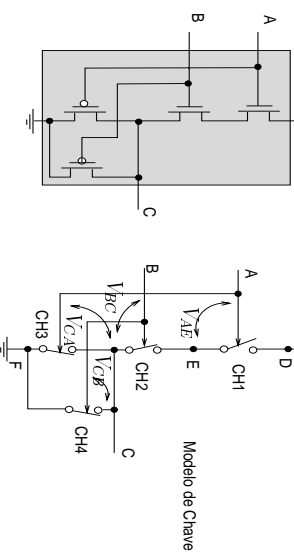
$$(a + b) = ((a + b)')'$$



Exemplos de implementações de funções mais complexas (seção 3.2.5 do livro-texto):



Pergunta 1: Podemos implementar uma porta AND com a seguinte conexão de transistores CMOS?

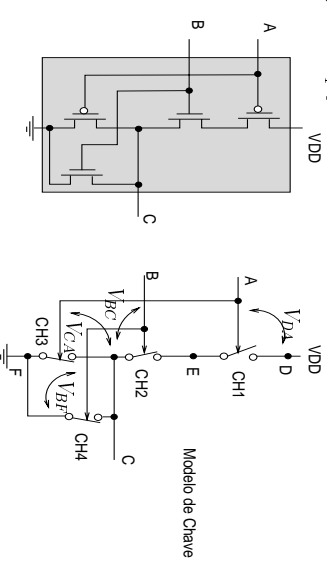


Vamos analisar os estados das chaves para cada possível combinação de entradas:

| A | B | CH1 | CH2 | CH3 | CH4 | C |
|----------------|----------------|-----|-----|-----|-----|---|
| 0 V (0) | 0 V (0) | ? | ? | ? | ? | ? |
| 0 V (0) | V_{DD} V (1) | ? | ? | ? | ? | ? |
| V_{DD} V (1) | 0 V (0) | ? | ? | ? | ? | ? |
| V_{DD} V (1) | V_{DD} V (1) | ? | ? | ? | ? | ? |

Notem que a tensão nos pontos C e E são indefinidos e não se pode dizer nada em relação aos estados das chaves. Portanto, a tensão de saída (e consequentemente, o seu nível lógico) não é definida para nenhuma combinação de entradas.

Pergunta 2: A seguinte conexão de transistores CMOS realiza a função lógica $z = x_1x_0$?



Vamos analisar os estados das chaves para cada possível combinação de entradas:

| A | B | CH1 | CH2 | CH3 | CH4 | C |
|-----------|-----------|---------|-----|-----|---------|---------|
| 0 V (0) | 0 V (0) | fechada | ? | ? | aberta | ? |
| 0 V (0) | VDD V (1) | fechada | ? | ? | aberta | ? |
| VDD V (1) | 0 V (0) | aberta | ? | ? | fechada | 0 V (0) |
| VDD V (1) | VDD V (1) | aberta | ? | ? | fechada | 0 V (0) |

Notem que a tensão no ponto C é indefinida e não se pode dizer nada em relação aos estados das chaves CH2 e CH3. Portanto, a tensão de saída (e consequentemente, o seu nível lógico) não é definida para as duas primeiras combinações de entradas.

Pergunta 3: Dada uma tabela-verdade, como

| A | B | C |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

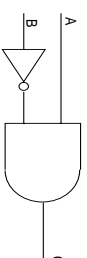
Como se pode implementar a lógica definida por ela?

A função de chaveamento correspondente é:

$$C = f(A, B) = \sum_{AB} (2) = AB'$$

Vimos até agora duas formas para implementar esta função:

1. **Com uso de portas lógicas:** Suponhamos que dispomos de portas AND e NOT, então



ou então só temos portaa NAND. Neste caso, com algumas manipulações algébricas chegamos a seguinte expressão

$$C = AB' = (AB)' = ((AB)'(AB))' = ((A(BB))'(A(BB)))'$$

que pode ser implementada com 3 portas NAND.

2. **Com uso de transistores CMOS:** Neste caso se reservemos a expressão em

$$C = AB' = (AB)'' = (A' + B)'$$

podamos compor o circuito de NOT com o de NOR para realizar a função

