

[View on GitHub](#)

Circuitos Digitais

Repo of Digital Circuits course - CRT0384

PRÁTICA 06 - CIRCUITOS LÓGICOS SEQUENCIAIS - LATCHES - SR e D

[Voltar à home](#)

OBJETIVOS

- Observar e analisar os princípios de funcionamento dos Flip-flop's e compreender o processo de armazenamento de informações utilizando portas lógicas.

Material Necessário:

- 02 TTL 74LS00;
- 01 TTL 74LS02;
- 01 TTL 74LS283

Vimos nas últimas práticas o funcionamento da lógica combinacional, ou ainda, uma lógica para a qual a saída é pré-determinada de forma unívoca pelas entradas. Estes circuitos, no entanto, não apresentam qualquer presença de elemento que possa armazenar informações lógicas de entrada, ou seja, memória sobre os estados anteriores. Todo o desenvolvimento computacional e de comunicações digitais está suportado pela concepção de memória.

Circuitos lógicos sequenciais são categorias de circuitos que, utilizando portas lógicas, implementam sistemas de memória ao ponto de que, em geral, a saída do sistema depende do estado atual e de estados anteriores. O principal elemento de memória utilizado é o *Flip-Flop*, constituído por portas lógicas que, sozinhas, não têm capacidade de armazenamento, mas conectadas entre si transformam o circuito em um sistema dotado de memória.

Os elementos fundamentais para construção de *flip-flops* são chamados *latches*, circuitos sequenciais com capacidade de armazenamento, que serão estudados hoje.

PARTE 1 - Latch tipo Set e Reset - SR

O *Latch* é um dispositivo de armazenamento temporário que tem dois estados estáveis (biestável). O *Latch* SR pode ser construído com portas NAND, conforme figura a seguir. Monte o circuito indicado e preencha a tabela ao lado:

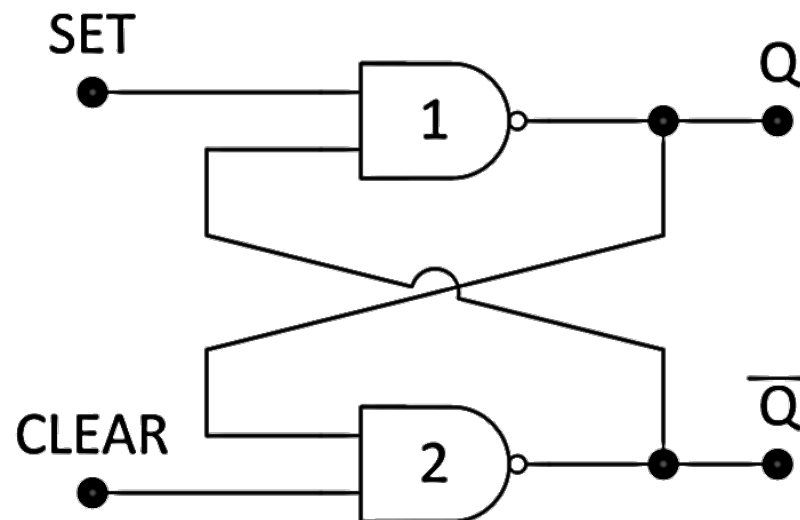


Tabela 1

SET	CLEAR	Q	Q'

Construa o Latch SR com portas NOR e preencha a tabela ao lado:

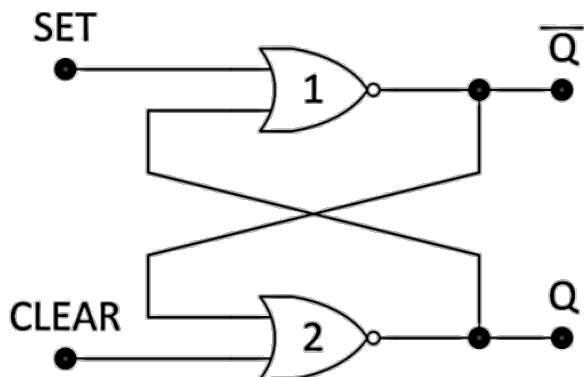
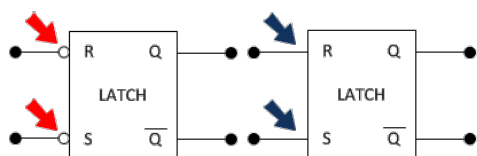


Tabela 1

SET	CLEAR	Q	Q'

Observe, a partir das tabelas-verdade, que as entradas do *latch* com portas NAND são ativas em nível baixo e as do *latch* com portas NOR são ativas em nível alto. Podemos representar os dois tipos de latch conforme descrito a seguir:



Latch NAND e Latch NOR, respectivamente

Verifique a relação das saídas Q e Q', para os dois modelos de latch SR, para um conjunto maior de configuração de entrada, como mostrado a seguir:

SET CLEAR Q ——— 0 0 0 1
 0 1 1 1
 1 0 0 0
 1 1 0 1
 1 0 1 1
 0 0 1 0
 1 0 0 0
 1 1 1 0

Verifique se são notadas as condições mostradas a seguir:

LATCH NAND LATCH NOR

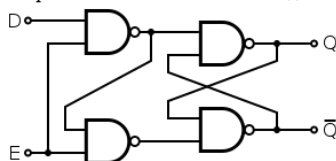
Set	Reset	Saída	Set	Reset	Saída
1	1	Não muda	0	0	Não muda
0	1	Q = 1	1	0	Q = 1
1	0	Q = 0	0	1	Q = 0
0	0	Inválida*	1	1	Inválida*

*Produz $Q = \bar{Q} = 1$.

*Produz $Q = \bar{Q} = 0$.

PARTE 2 - Flip-Flop tipo D

Latch tipo D tem uma entrada chamada D (dados) e duas saídas correspondentes ao estado (Q) e seu complemento (Q'). Uma entrada adicional chamada EN. Neste latch, o próximo estado Q(t+1) corresponde ao valor da entrada D(t). Dessa forma, o latch D é chamado latch de "retardo" ou



"transparente". O diagrama lógico é exibido a seguir:

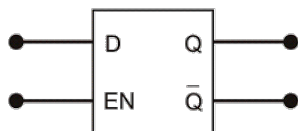
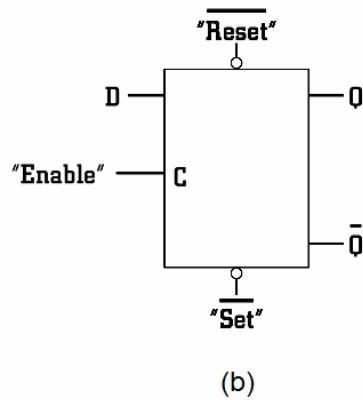


Figura 4: Latch tipo D

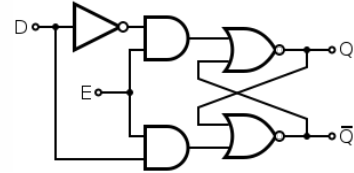
Latch tipo D com portas NAND

"Enable"	D	Q
0	X	Não muda
1	1	1
1	0	0

(a)



(b)



Latch tipo D com portas NOR, AND e NOT

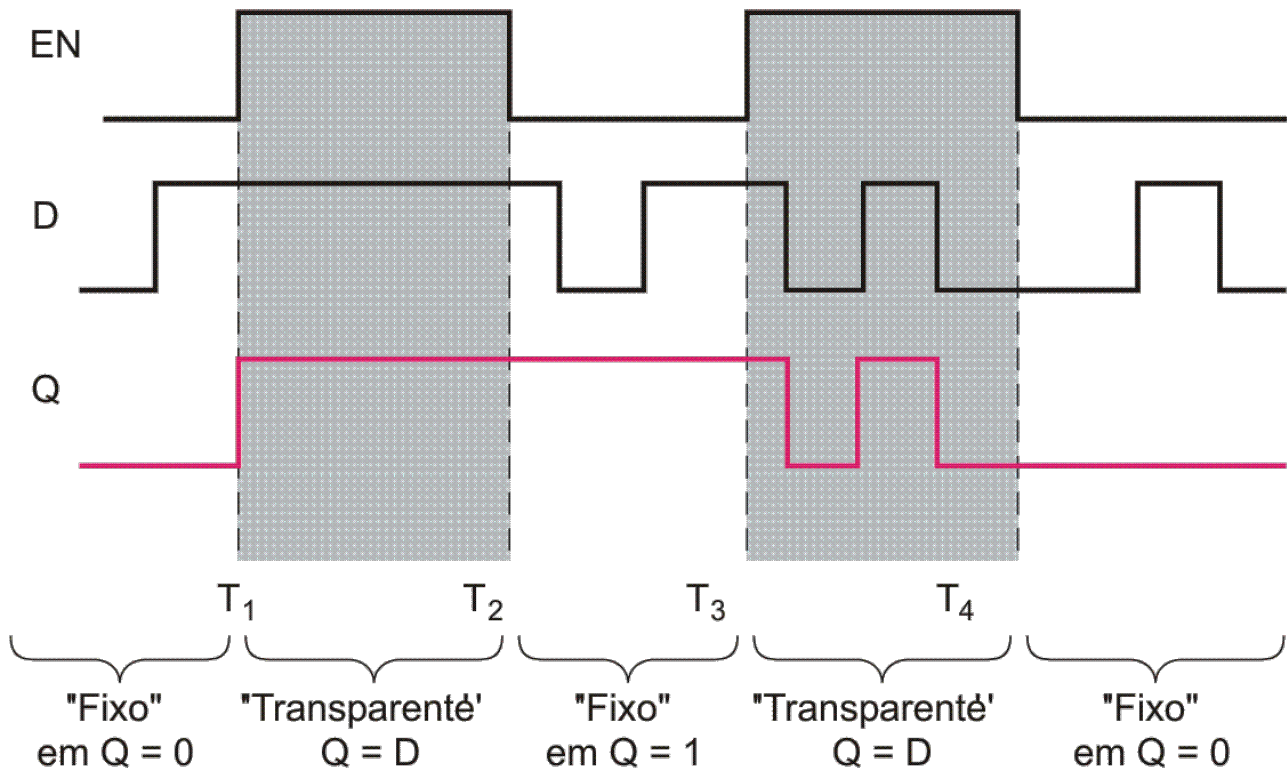
O latch D pode ser construído a partir do latch SR conforme diagrama lógico acima. Implemente um latch tipo D utilizando portas **NAND** e preencha a tabela verdade a seguir:

DATA ENABLE Q ————— 0 0 0 1
 0 1 1 1
 1 0 0 0
 1 1 0 1
 1 0 1 1
 0 0 1 0
 1 0 0 0
 1 1 1 0

Implemente também um segundo latch NAND, conforme ilustrado. Preencha a tabela novamente

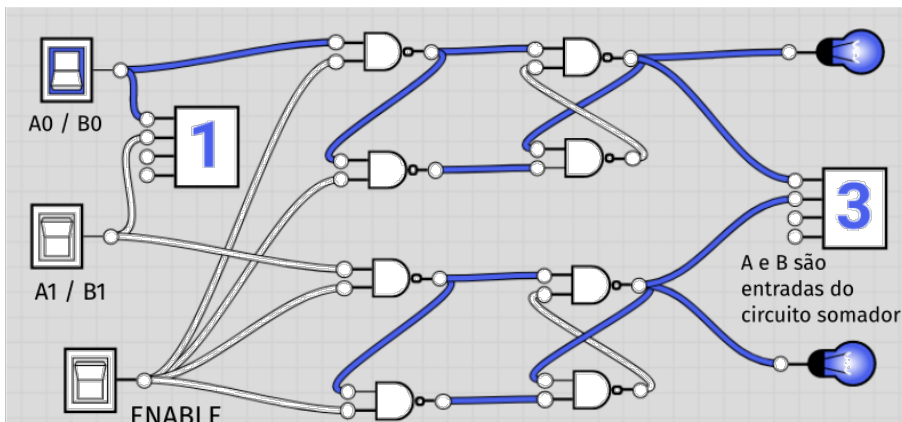
DATA ENABLE Q ————— 0 0 0 1
 0 1 1 1
 1 0 0 0
 1 1 0 1
 1 0 1 1
 0 0 1 0
 1 0 0 0
 1 1 1 0

Exemplo do comportamento de um latch D para as formas de onda dadas:

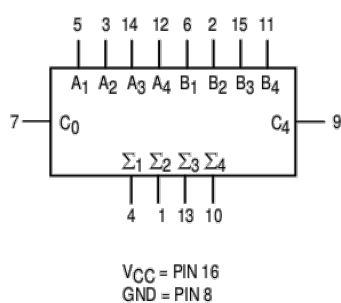
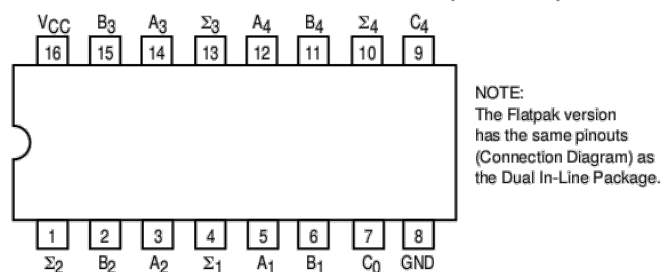


PARTE 3 - CIRCUITOS DIGITAIS COM MEMÓRIA:

Sem desmontar os latches tipo D, implemente um somador de dois bits usando o CI 74LS283. O somador deverá ser acionado por dois bits de saída dos latches D e por duas entradas das chaves. Use as mesmas chaves para inserir dados para os *latches* e para o somador, usando uma chave extra de ENABLE para habilitar a gravação nos *latches* ou não. O diagrama é representado a seguir:

**PÓS LABORATÓRIO - RELATÓRIO:**

1. Faça o detalhamento, no relatório, dos procedimentos relativos à montagem dos circuitos da prática. Exiba a tabela de saída obtida dos circuitos.
2. Descreva a importância do uso de *latches* e *flip-flops* na implementação de circuitos que possuem memória. Quais são as aplicações dos *latches* e *flip-flops*?
3. Pesquise sobre memórias estáticas de acesso aleatório - SRAM. Como essas memórias podem ser implementadas utilizando circuitos lógicos?
4. Implemente um circuito lógico que equivalha a um latch tipo D utilizando APENAS portas NOR;
5. Implemente um circuito somador que receba duas entradas sequenciais de quatro bits, de modo que as chaves de inserção sejam compartilhadas para as entradas A e B, ou seja, usando latches tipo D para armazenagem.

DIAGRAMAS:**LOGIC SYMBOL****CONNECTION DIAGRAM DIP (TOP VIEW)**

PIN NAMES

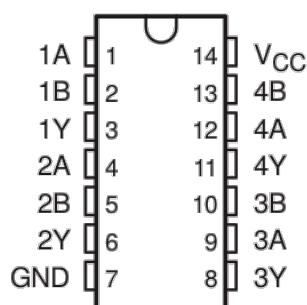
$A_1 - A_4$ Operand A Inputs
 $B_1 - B_4$ Operand B Inputs
 C_0 Carry Input
 $\Sigma_1 - \Sigma_4$ Sum Outputs (Note b)
 C_4 Carry Output (Note b)

CIs Lógicos:

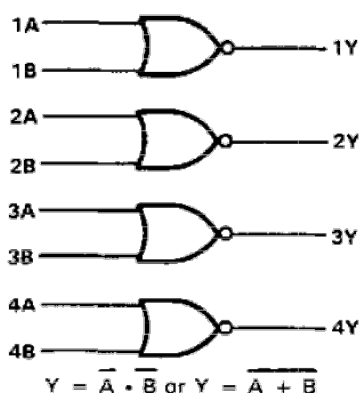
7400 - NAND



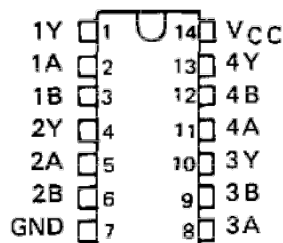
SN5400 . . . J PACKAGE
SN54LS00, SN54S00 . . . J OR W PACKAGE
SN7400, SN74S00 . . . D, N, OR NS PACKAGE
SN74LS00 . . . D, DB, N, OR NS PACKAGE
(TOP VIEW)



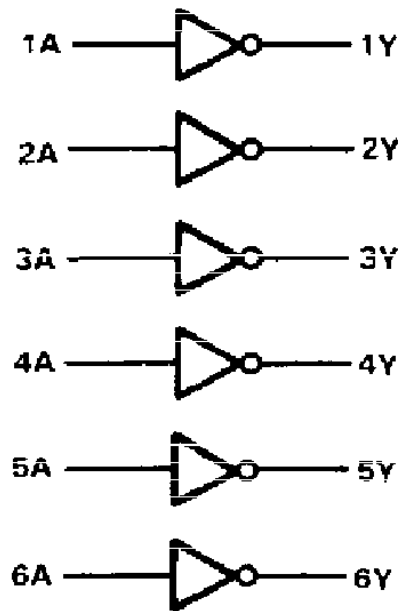
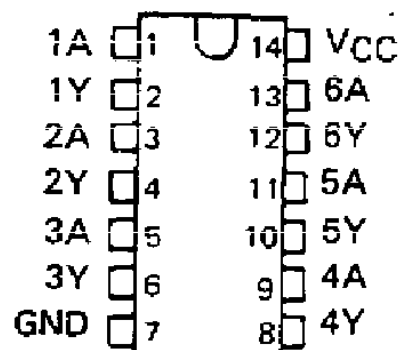
7402 - NOR

logic diagram (positive logic)

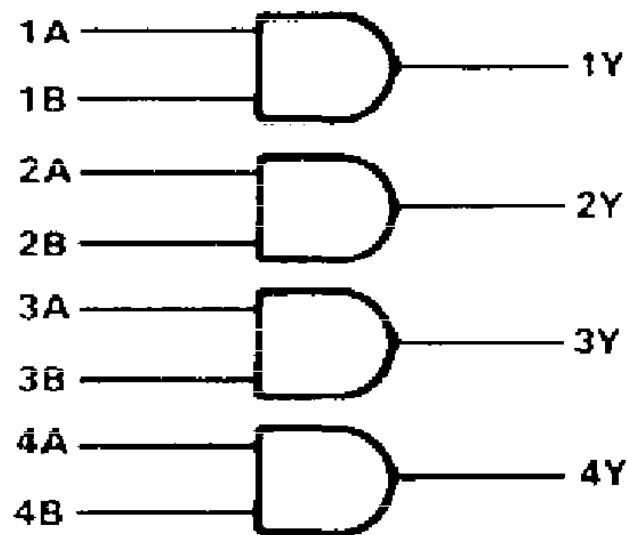
SN5402 . . . J PACKAGE
SN54LS02, SN54S02 . . . J OR W PACKAGE
SN7402 . . . N PACKAGE
SN74LS02, SN74S02 . . . D OR N PACKAGE
(TOP VIEW)



7404 - NOT

logic diagram (positive logic)**SN5404 . . . J PACKAGE****SN54LS04, SN54S04 . . . J OR W PACKAGE****SN7404 . . . N PACKAGE****SN74LS04, SN74S04 . . . D OR N PACKAGE****(TOP VIEW)**

7408 - AND

logic diagram (positive logic)

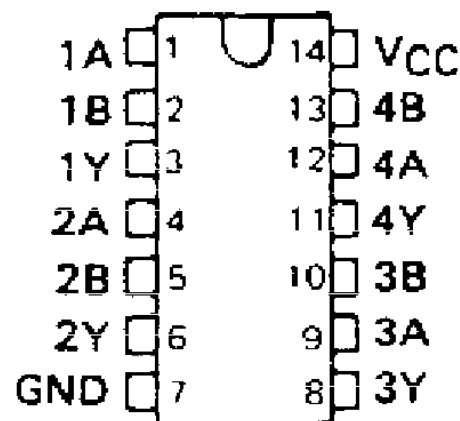
$$Y = A \cdot B \text{ or } Y = \overline{\overline{A} + \overline{B}}$$

SN5408, SN54LS08, SN54S08 . . . J OR W PACKAGE

SN7408 . . . J OR N PACKAGE

SN74LS08, SN74S08 . . . D, J OR N PACKAGE

(TOP VIEW)



Circuitos Digitais maintained by [marcielbp](#)

Published with [GitHub Pages](#)