Arquitetura de computadores I

Formas de representação

Min

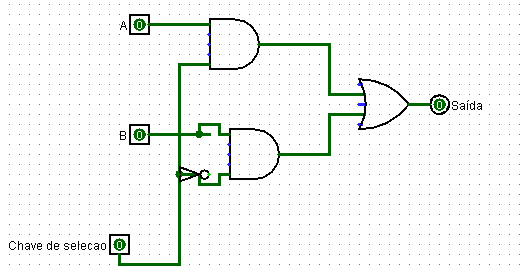
Circuitos BÁsicos

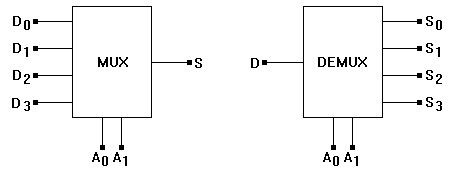
Multiplexadores

Também chamados de seletores de dados são circuitos lógicos capazes de atuar como chaves digitais, recebem uma entrada, selecionam uma delas e transferem para a saída. Podem ser usados para rotear dados, sequenciar operações, realizar conversões do tipo paralelo-série, entre outras.

Resumindo: duas entradas, uma seleção, uma saída,

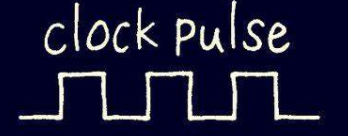
Um **demux** é o circuito lógico que vai fazer o processo inverso do **mux** ele recebe uma entrada e distribui os dados entre várias saídas. Podem ser usados para distribuir o mesmo sinal de ativação, ou sequenciamento clock para vários circuitos





**Clock**: do inglês relógio, é um **sinal digital pulsante** que mantém os circuitos em sincronia, para manter tudo trabalhando ao mesmo tempo e a uma mesma velocidade. O sinal gera uma certa frequência (quantidade de pulsos por segundo), medida em hz.

O Duty cicle é o a relação entre o tempo ativo e o período do clock. Ht/TT



Máquinas de estado finito

Nos ajudam a reconhecer algumas coisas, tal como um efeito de memória. Podemos representá-los através de grafos que compõem entrada e saída.

**Usos**:

Reconhecedores.

Classificadores.

Transdutores.

Sequenciadores.

Autômato finito

É o gol de ouro, a primeira vez que tivermos o que queremos acabou.

Mealy

São autômatos finitos que gravam símbolos de saída, o seja as saída dependem do estado corrente e das entradas. Pode–se dizer que é um autômato ufanista.

Moore

Modelo que depende apenas do estado corrente. É o pessimista.

Máquina de Turing

Modelo matemático computacional que manipula símbolos em uma fita seguindo certas regras.

Autômatos de Pilha

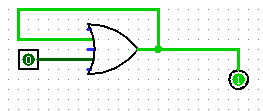
Também chamado de pushdown automata é um modelo matemático composto por um símbolo de entrada, um símbolo de topo e um símbolo de inserção. Esse modelo compara a entrada com o topo da pilha, dependendo do resultado guardará a saída no topo da pilha e prosseguirá para o próximo estado.

**Armazenamento de memória**

**Circuitos combinacionais e sequenciais**

Em **circuitos combinacionais** a saída depende apenas e somente do estado lógico das entradas,ou seja, não possuem memória. Já em **circuitos sequenciais** ela dependa do estado das entradas e do estado anterior da própria saída, ou seja, temos o efeito de memória.

A memorização acontece quando fazemos a realimentação da saída no circuito.



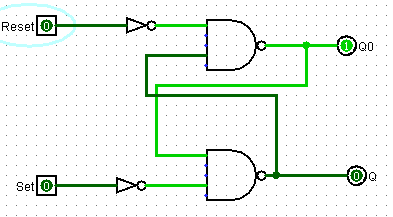
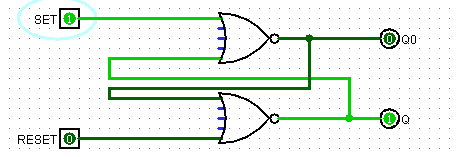
***Existindo clock temos um flip flop, não existindo teremos uma latch.***

Latch RS

Uma **latch** é uma estrutura lógica capaz de armazenar um bit. Constituída por um circuito que muda de estado apenas devido às variações de entrada. Muito utilizadas para construir chaveadores sem ressalto.

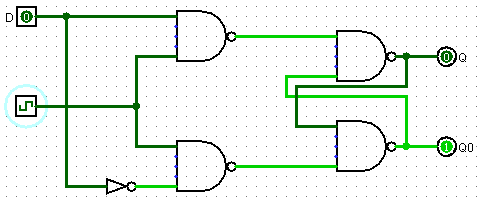
Latch Rs

**Latch SR com nands Latch Sr com nors**

Latch D

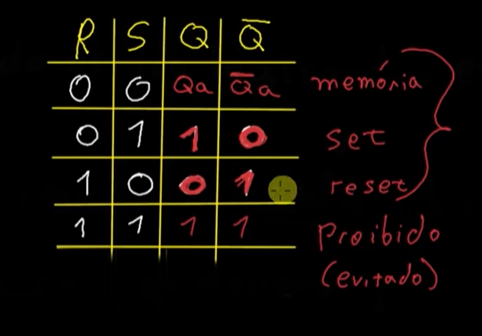
Um latch D mantém a mantagem de fazer um hold, mesmo que desabilitado, além disso, é capaz de impedir a ocorrência do estado proibido por vincular entradas (S e R) em uma única (D) e sua inversão.



Contudo a latch D possui uma desvantagem, se o sinal de controle for mantido em 1 o e sinal de entrada flutuar, as saídas também irão. O latch D não reconhece estabilidade.

**Flip Flops**

Circuito digital com pelo menos duas entradas, uma de acionamento, uma para desligar, e um estado de memorização. Sendo R o reset, S de set e uma saída Q e ‘Q. As transições de estado ocorrem somente durante as transições (positiva de subida ou negativa de descida) de um clock.



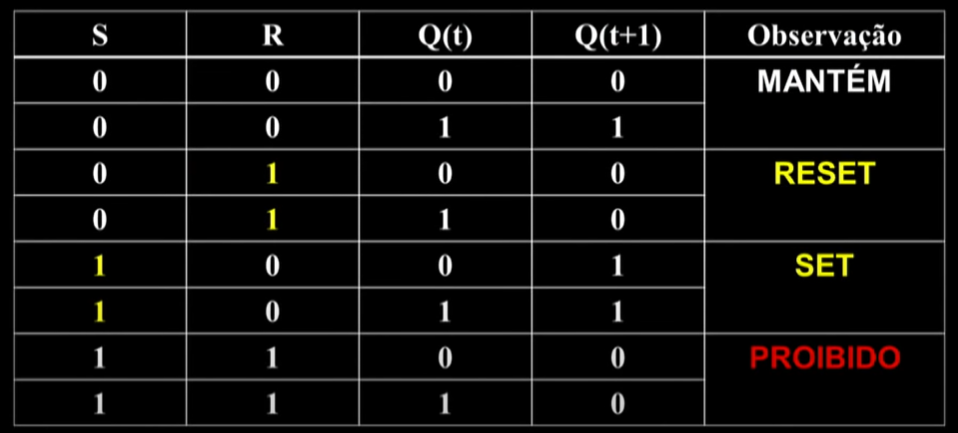
**Resumindo**

Hold - quando mantemos o valor. 0 0 0.

Reset – Zera a saída, em resumo desliga. q1= 0, Q0 = 1.

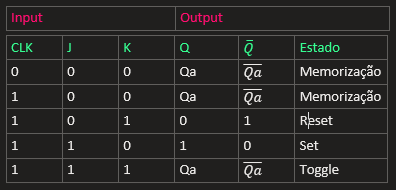
Set – Liga a saída Q0 = 1.

Unset – é igual a X.



Flip flop JK

Evolução do flip flop RS e seu objetivo é eliminar a condição proibida (duas saídas de mesmo estado). Tem duas entradas, um clock e duas saídas. Quando forem iguais o flip flop faz um toggle, invertendo as entradas fazendo uma oscilação. É considerado um tipo universal para se construir outros tipos.



**Resumindo:** J de jump (ou jóia) e K de kill ou (Karalho fudeu), evolução do Rs se tivermos j=1 e k=1ele faz o **toggle**.

**Contadores**

Contadores são circuitos que conseguem contar o número de pulsos recebidos, o **módulo** de um contador é o número de estados que o contador possui, ou seja um contador de 1 bit conta até 1 um de 2 bits conta até 2 e assim por diante.

Contadores crescentes contam de forma a somar os números, em contrapartida, os decrescentes subtraem.

**Contadores assíncronos**: os pinos de clock dos flip flops é separado, ou seja o clock vai no primeiro e após isso o segundo muda de estado. São mais simples de se projetar e bem mais lentos.

Crescente:

**Contadores síncronos**: são aqueles que possuem os pinos de clock em paralelo, ou seja o clock vai para todos ao mesmo tempo, devido à isso são bem mais rápidos que os assíncronos e mais complexos. Além disso podem implementar sequências além das crescentes e decrescentes.

**Memórias**

**Memória Rom**

Memória reservada apenas para leitura.

**Memória Ram**

Memória rápida de acesso randômico

**Registradores**

São associações de 2 ou mais flip flops que controlados por um clock comum onde, cada armazena um bit, podendo existir entradas síncronas ou assíncronas, e as saídas de alguns flip flops podem também estar ligadas.

**Em resumo**, servem para armazenar um conjunto de bits.

Sinais de controle:

-Enable.

-Set(preset).

-Reset.

-Shift.

-Load.

Registradores de saída em série (Shift - register)

**Memória de registradores**

Memória que guarda trabalho