

## REGISTRADORES

Os flip-flops podem ser agrupados para formar circuitos isolados com uma aplicação específica e limitada, chamados de subsistemas seqüenciais. Juntos, os subsistemas formam sistemas maiores, como o computador.

Três subsistemas seqüenciais fundamentais são: os registradores, os contadores e as memórias. Registrador é um subsistema seqüencial constituído basicamente por flip-flops, e serve para a manipulação e armazenamento de dados. Para nosso estudo, entende-se como dados a informação no formato binário (na forma de bits).

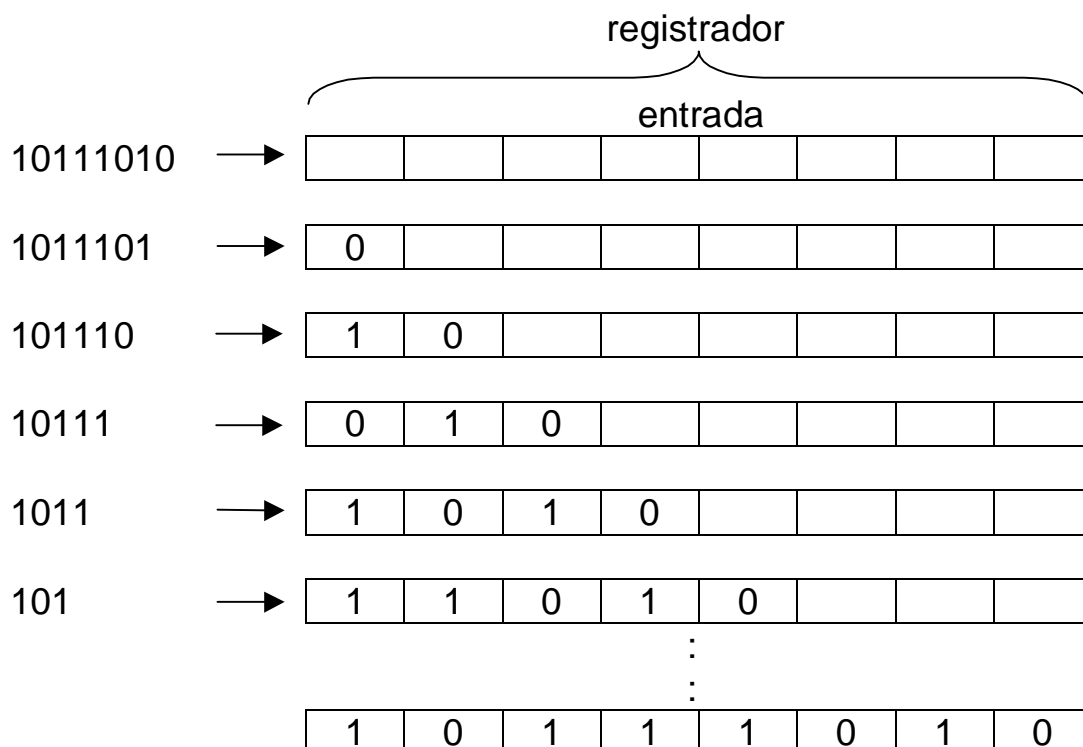
O registrador possui quatro configurações diferentes, dadas pelo modo como os dados (bits) entram e como são transmitidos do registrador para outro circuito.

### Configurações básicas

- Modo serial: os dados são recebidos e/ou transmitidos um bit por vez, em uma única linha;
- Modo paralelo: mais de um bit que compõem os dados são recebidos e/ou transmitidos simultaneamente, em mais de uma linha.

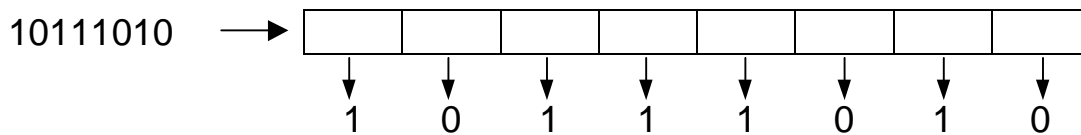
De acordo com os modos de entrada e saída dos bits, os registradores podem ser classificados como:

#### 1. Registrador série-série:

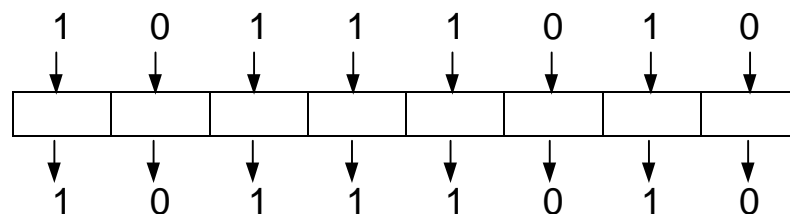




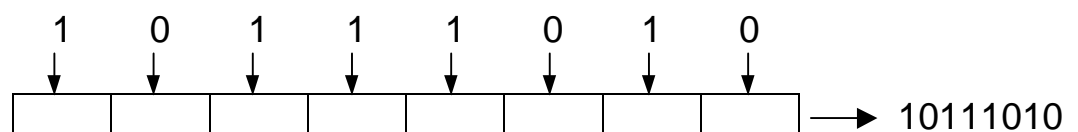
## 2. Registrador série-paralelo:



## 3. Registrador paralelo-paralelo:



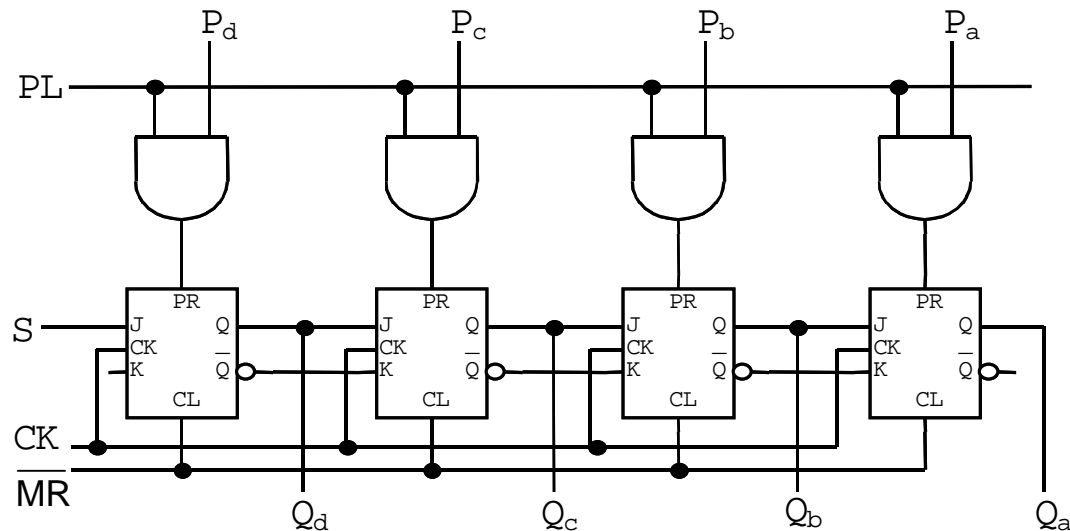
## 4. Registrador paralelo-série:



O número de bits que um registrador pode armazenar é igual ao número de flip-flops que o compõe. Nas diferenças entre o modo serial e o modo paralelo podemos notar que o modo paralelo fornece simultaneamente a entrada e/ou saída dos bits, sendo mais rápido, porém exige mais linhas de entrada e/ou saída (mais circuitos ligados ao registrador). No modo serial os bits deslocam-se em sincronia, sendo mais lento porque no intervalo de tempo em que um bit é deslocado no modo

paralelo pode ser deslocado mais de um bit, mas o serial exige somente um circuito ligado à entrada e/ou saída do registrador.

### Registrador de Deslocamento (Shift Register)



São registradores que utilizam entrada e/ou saída seriais e os dados movimentam-se internamente, sendo chamados também de registradores de deslocamento (shift registers).

O funcionamento deste registrador depende da forma como são ativadas as entradas CK,  $\overline{\text{MR}}$ , PL e S:

- $P_a$ ,  $P_b$ ,  $P_c$  e  $P_d$  são as entradas paralelas;
- $Q_a$ ,  $Q_b$ ,  $Q_c$  e  $Q_d$  são as saídas paralelas,  $Q_a$  também é a saída serial;
- CK – entrada do pulso de clock: possibilita o deslocamento dos dados do registrador;
- MR – entrada master reset: habilita as entradas clear ( $\overline{\text{CL}}$ ) de todos os flip-flops, fazendo com que as saídas  $Q_d$ ,  $Q_c$ ,  $Q_b$  e  $Q_a$  fiquem resetadas (em nível lógico “0”);
- PL - entrada paralela (parallel load): habilita as entradas paralelas transferindo-as para as saídas paralelas;
- S – entrada serial: por onde os dados entram serialmente para serem armazenados no registrador.

As entradas PR (PRESET) se habilitadas (no diagrama acima, se PR=“1”) levam a saída do respectivo flip-flop para “1”, se desabilitadas (PR=“0”) não mudam o valor da saída no flip-flop. As entradas CL (CLEAR) se habilitadas (CL=1) levam a saída à “0”, se desabilitadas (CL=0) não mudam o valor da saída do flip-flop. Observar que para habilitar as entradas CL de cada flip-flop existe uma única entrada  $\overline{\text{MR}}$ , que está invertida: para habilitar CL de cada flip-flop deve-se aplicar o sinal “0”, para

evitar a confusão e a troca de sinais. Assim, o sinal de nível lógico “0” limpa, ou zera a saída de todos os flip-flops.

Este registrador é denominado síncrono porque os pulsos de clock ativam todos os flip-flops simultaneamente. Porém as entradas  $\overline{MR}$  e PL são denominadas assíncronas pois independem do pulso de clock. Os modos de funcionamento do registrador de deslocamento estão descritos a seguir:

#### 1. Série-série:

Neste modo e nos seguintes, os quais exigem entrada serial, a entrada paralela PL é desabilitada (no diagrama acima  $PL=0$ ) e os bits entram serialmente (um por vez) pela entrada S. A cada pulso do clock os bits passam para a saída do flip-flop e para a entrada do próximo flip-flop logo à direita, ou seja, deslocam-se para a direita e são transmitidos a outro circuito por  $Q_a$ . O primeiro bit que se deseja obter em  $Q_a$  é o primeiro bit que entra por J. Então, por exemplo, os bits 1101 são armazenados no registrador série-série na ordem 1101, para sair em  $Q_a$  na ordem em que entraram por S.

Caso se deseje, pode-se habilitar o reset do registrador ( $\overline{MR}=0$ ) e desabilitá-lo em seguida ( $\overline{MR}=1$ ), zerando todas as saídas dos flip-flops antes da entrada de novos dados.

#### 2. Série-paralelo:

Procede-se de modo igual ao registrador série-série, com a diferença de que as saídas paralelas são obtidas diretamente das saídas  $Q_a$  a  $Q_d$ . Para isto, deve-se aguardar o último pulso de clock para que a saída do último flip-flop do registrador seja atualizada com o último bit do dado de entrada, para então efetuar a transmissão paralela.

#### 3. Paralelo-paralelo:

O registrador é resetado pela habilitação do master reset ( $\overline{MR}=0$ ) que logo após é desabilitado ( $\overline{MR}=1$ ). Observar que as saídas do flip-flop devem ser resetadas sempre que a entrada paralela for habilitada ( $PL=1$ ), evitando erros nas saídas dos flip-flops. Deste modo os bits em  $P_a$ ,  $P_b$ ,  $P_c$  e  $P_d$  são enviados às saídas dos flip-flops caso os valores sejam “1”. Caso alguns bits em  $P_a$  a  $P_d$  sejam “0”, então as respectivas saídas  $Q_a$  a  $Q_d$  não são modificadas, e como foram resetadas, permanecem em “0”. A transmissão do dado se faz da mesma forma que no modo anterior.

#### 4. Paralelo-série:

Neste modo a entrada se procede da mesma forma que no modo anterior, porém a transmissão dos dados se faz de forma serial, dependente do sinal de clock, para que os bits saiam por  $Q_a$ . Assim, a entrada de dados paralelamente só pode ser

feita após o último bit do dado ser transmitido (deslocado) por  $Q_a$ . Logo após, o registrador deve ser resetado e os novos dados podem entrar paralelamente.