

Registadores

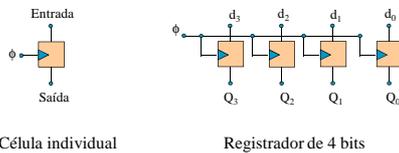
Aula 17
Prof. Abel Guilhermino

Registrador

- Um registrador é um elemento lógico utilizado para armazenar uma palavra binária de n-bits.
- Praticamente, todos os grandes sistemas digitais utilizam registradores para armazenar dados importantes.

Registrador básico

- Pode-se construir um registrador com a conexão de n células de armazenamento, de um único bit, em paralelo, possibilitando ler ou escrever em todas as células simultaneamente.
- Vamos examinar um registrador de 4 bits que pode armazenar e bits de dado.



Exemplo (registrador de 8 bits)

- Seu código em Verilog:

```
module regis (clk,din,dout);  
  input clk;  
  input [7:0] din;  
  output [7:0] dout;  
  reg [7:0] dout;  
  
  always @(posedge clk)  
  begin  
    dout <= din;  
  end  
  
endmodule
```

Obs:
Basta identificar uma borda de subida do sinal de clock para que a entrada *din* seja atribuída à saída *dout*.

Exemplo (registrador de 8 bits)

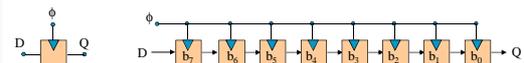
- Seu código em VHDL:

```
entity reg is  
  port ( clk : in bit;  
        din : in bit_vector(7 downto 0);  
        dout : out bit_vector(7 downto 0) );  
end reg;  
architecture arch of reg is  
  begin  
    process (clk)  
      begin  
        if (clk'event and clk='1') then  
          dout <= din;  
        end if;  
      end process;  
end arch;
```

Obs:
Basta identificar uma borda de subida do sinal de clock para que a entrada *din* seja atribuída à saída *dout*.

Registrador de Deslocamento

- Um registrador de deslocamento é projetado para mover bits para as células vizinhas, enquanto houver pulsos de clock.



- Como todas as células são controladas pelo mesmo sinal de clock ϕ , todas elas são carregadas ao mesmo tempo

Transferência de informação

Transferência em paralelo



Transferência serial

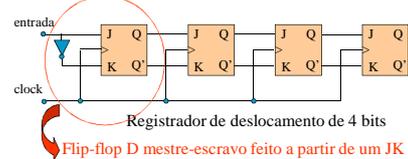


7

Registrador de Deslocament

J	K	Q
0	0	Q _n
0	1	0
1	0	1
1	1	Q _n '

- Como sabemos o flip-flop pode armazenar um estado de saída, até que um novo pulso chegue ao clock.
- Registrador de deslocamento trata-se de um certo número de flip-flops RS ou JK mestre-escravo, sendo cada J ou S ligada à saída Q do flip-flop anterior e cada entrada K ou R ligada à saída Q' do flip-flop anterior.
- O primeiro flip-flop é ligado com um flip-flop tipo D



Conversor Série-Paralelo

- Informação Paralela : os bits se apresentam simultaneamente.

- Apresenta tantos fios quanto o número de bits
- Caso: Queremos transmitir a informação

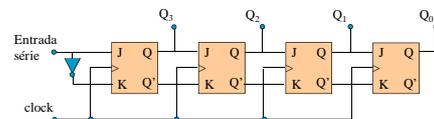
$$I_3 I_2 I_1 I_0 = 1010$$



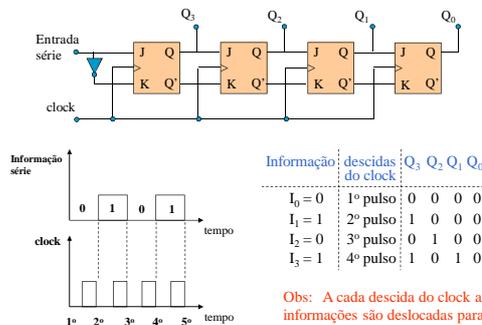
Conversor Série-Paralelo

- Informação Serial : os bits são transmitidos por um único fio.

- Podemos construir um conversor série-paralelo com um registrador de deslocamento



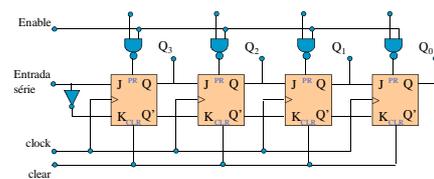
Conversor Série-Paralelo



Conversor Paralelo-Série

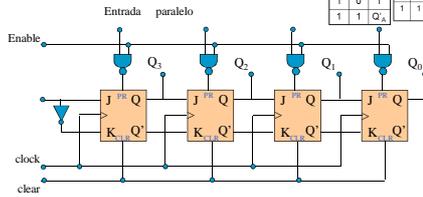
J	K	Q	CL	PR	Q
0	0	Q _n	0	0	Não permitido
0	1	0	0	1	0
1	0	1	1	0	1
1	1	Q _n '	1	1	Funcionamento normal

- Podemos realizar o processo inverso, ou seja, entrar com uma palavra com n bits e "retirar" bit a bit com pulsos de clock.
- Para isto utilizamos as funções Preset e Clear dos flip-flops JK mestre-escravo.
- O circuito utilizado está mostrado abaixo:



Conversor Paralelo-Sér

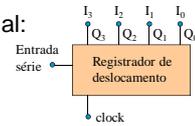
J	K	Q	CL	PR	Q
0	0	0	0	0	Não permitido
0	1	0	0	1	0
1	0	1	1	0	1
1	1	Q _n	1	1	Funcionamento normal



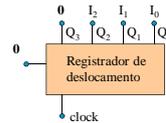
- Para inserirmos a palavra paralela:
 - Primeiro damos um pulso em zero no Clear e todas as saídas vão para estado zero (Enable = 0 => Prset = 1).
 - Contudo, enquanto o enable estiver em zero, a entrada preset estará em '1', e as saídas permanecerão em zero.
 - Quando a entrada ENABLE for para '1', as saídas assumirão os valores presentes nas entradas PRESET, ou seja, quando ENABLE='1' → Q₃ = \PR3; Q₂ = \PR2; Q₁ = \PR1; Q₀ = \PR0
 - Então para externar serialmente os bits, aplica-se pulsos ao clock.

Divisor por 2

- Considere o estado inicial:



- Agora considere a entrada série em zero e um pulso de clock aplicado. Assim, temos:



Ou seja, o número na saída é uma potência de 2 menor.

Exemplo: (divisão por 2)

- Considere: I₃I₂I₁I₀ = (1010)₂ = (10)₁₀

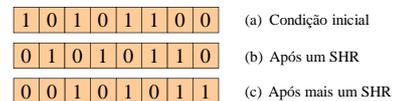
Depois do pulso de clock teremos:

$$0I_2I_1I_0 = 0101 = (5)_{10}$$



Outros registradores

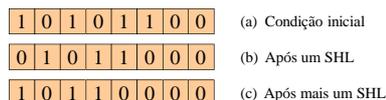
- Um registrador de deslocamento de carregamento paralelo permite ao usuário armazenar toda a palavra em uma transição de clock.
- Este registrador pode possuir duas operações de deslocamento controladas pelos sinais SHR (Shift Right) e SHL (Shift Left).
- Operação Shift Right:
 - A operação de deslocamento a direita é iniciada fazendo SHR=1.
 - Neste caso cada bit é movido uma posição para a direita
 - O bit mais a esquerda é forçado para o valor 0 (zero).



Obs: A operação SHR pode ser utilizada para dividir o conteúdo do registrador por uma potência de 2

Outros registradores

- Operação Shift Left:
 - A operação de deslocamento a direita é iniciada fazendo SHL=1.
 - Neste caso cada bit é movido uma posição para a esquerda
 - O bit mais a direita é forçado para o valor 0 (zero).

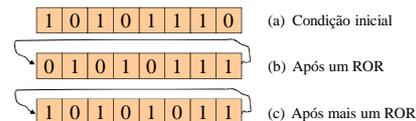


Obs: A operação SHL pode ser utilizada para multiplicar o conteúdo do registrador por uma potência de 2

Obs: As operações de divisão e multiplicação usando os operadores SHL e SHR só funciona corretamente para números decimais pares inteiros.

Outros registradores

- Operação ROR (Rotate Right)
 - Rotacionamento para a direita
 - Ao contrário do SHR o bit mais a direita é transferido para o outro lado do registrador.



Outros registradores

- Operação ROL (Rotate Left)
 - Rotacionamento para a esquerda
 - Ao contrário do SHL o bit mais a esquerda é transferido para o outro lado do registrador.

1 0 1 0 1 1 1 0 (a) Condição inicial

0 1 0 1 1 1 0 1 (b) Após um ROL

1 0 1 1 1 0 1 0 (c) Após mais um ROL

Exercícios

- Implementar através de componentes lógicos um registrador Shift Right de 4 bits usando flip-flops JK mestre-escravo.
- Implementar através de componentes lógicos um registrador Shift Left de 4 bits usando flip-flops JK mestre-escravo.
- Implementar através de componentes lógicos um registrador ROR de 4 bits usando flip-flops JK mestre-escravo.
- Implementar através de componentes lógicos um registrador ROL de 4 bits usando flip-flops JK mestre-escravo.

Exercícios

- Implementar através em Verilog um registrador Shift Right.
- Implementar através em Verilog um registrador Shift Left.
- Implementar através em Verilog um registrador ROR.
- Implementar através em Verilog um registrador ROL.
- Implemente um divisor por 4 usando flip-flops