



Laboratório 7

1. Considere seis elementos de memória como os dados em aula (*Latches* e *Flip-Flops*), com saídas Q1, Q2, Q3, Q4, Q5, e Q6. As suas entradas de dados estão ligadas aos sinais A e/ou B, dependendo do elemento ter uma ou duas entradas. Os componentes que geram as saídas **Q3**, **Q4** e **Q6** têm apenas uma entrada de dados e os demais (**Q1**, **Q2** e **Q5**) têm as duas entradas **A** e **B**. As entradas **m** e **n** forçam zeros ou uns. Outra entrada comum para a maior parte dos componentes é o sinal de **clk**.

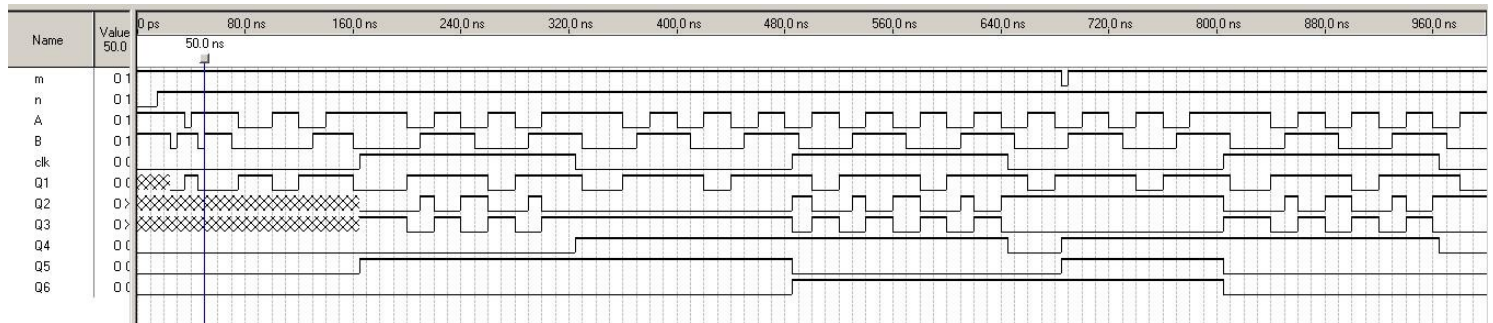


Figura 1: Forma de onda para diversos componente

A Figura 1 mostra formas de onda obtidas em uma simulação funcional.

- Determine quais sinais de saída, Q, correspondem às saídas de *latches* e quais correspondem às saídas de *Flip-Flops*.
- Determine quais componentes possuem *preset* ou *clear* síncrono ou assíncrono.
- Identifique os 6 componente .
- Implemente o Componente 1. [Q1.vhd]
- Implemente o Componente 2. [Q2.vhd]
- Implemente o Componente 3. [Q3.vhd]
- Implemente o Componente 4. [Q4.vhd]
- Implemente o Componente 5. [Q5.vhd]
- Implemente o Componente 6. [Q6.vhd]

Faça as simulações tentando reproduzir as formas de onda.

Observação: Para os seis componentes implementados, utilize os mesmos nomes de sinais da forma de onda. Nomeie as entidades de acordo com o arquivo a submeter.



2. Seja o registrador de n mostrado na Figura 2.

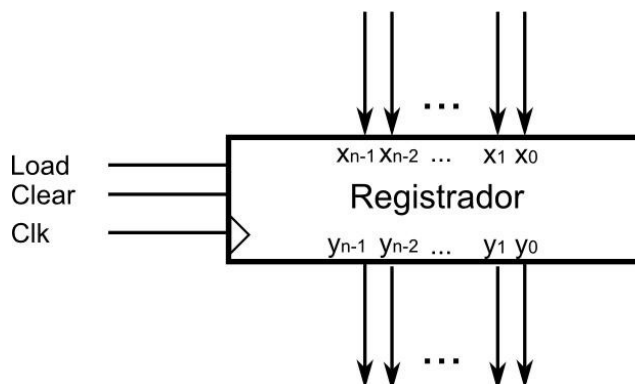


Figura 2: Registrador de n bits

- Implemente um registrador de n bits com carga síncrona e clear assíncrono. [RegN.vhd]
- Elabore uma forma de onda para testar o seu circuito.
- Implemente um decodificador 3x8 utilizando VHDL comportamental e salve-o para utilizar no circuito da Figura 3.
- Implemente o circuito da Figura 3 utilizando como componentes o registrador do item a, o decodificador do item c e o buffer tri-state dado em aula (ver buffer tri-state na aula sobre circuitos combinacionais típicos). Este banco tem 8 registradores de 4 bits e permite a escrita do registrador selecionado por SW(2..0) (D2R – Data to Register) com o valor na entrada DATA_IN(3..0). Também permite ler a qualquer instante o valor do registrador selecionado por SW(5..3) na saída DATA_OUT(3..0) (R2D – Register to Data). Esse banco de registradores será utilizado no m1ps. [Banco.zip, tendo Banco.vhd como top level entity]
- Instancie o projeto do item d no demo_setup, utilizando as 10 toggle switches como entrada de dados e seleção de registrador para ser escrito e lido (conforme a Figura 3). Utilize os push buttons para os sinais Clear, Load e Clock. Mostre o valor armazenado no Display de 7 segmentos hexadecimal. Verifique o funcionamento do reset assíncrono e do Load síncrono.

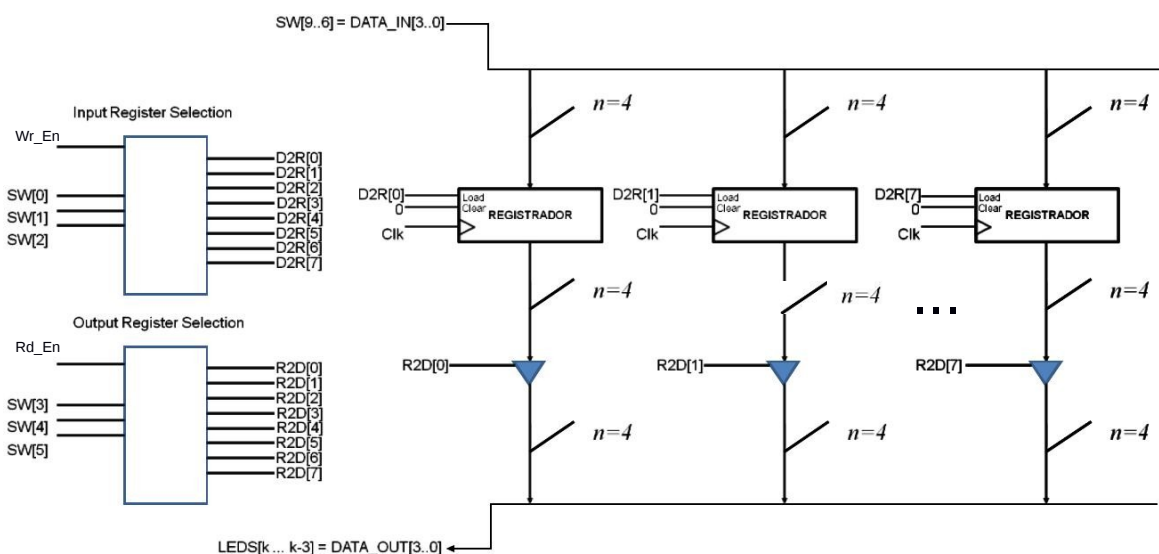


Figura 3: Banco de registradores (8 reg de 4 bits)

3. Usando os componentes fornecidos para o teclado PS/2 na página da tarefa, projete um leitor de teclado que capture o código da tecla pressionada por um teclado PS/2 ligado a placa (Obs.: Antes de fazer este exercício leia os tutoriais fornecidos na página da tarefa.)

a) Mostre o código da tecla pressionada no *display* de sete segmentos.

b) Faça um tradutor que seja capaz de mostrar o caractere pressionado no *display*, para caracteres alfanuméricos. Ignore os caracteres não alfanuméricos e não se preocupe com ambiguidades. (Esta tabela contém o mapeamento dos caracteres: <http://www.computer-engineering.org/ps2keyboard/scancodes2.html>)