

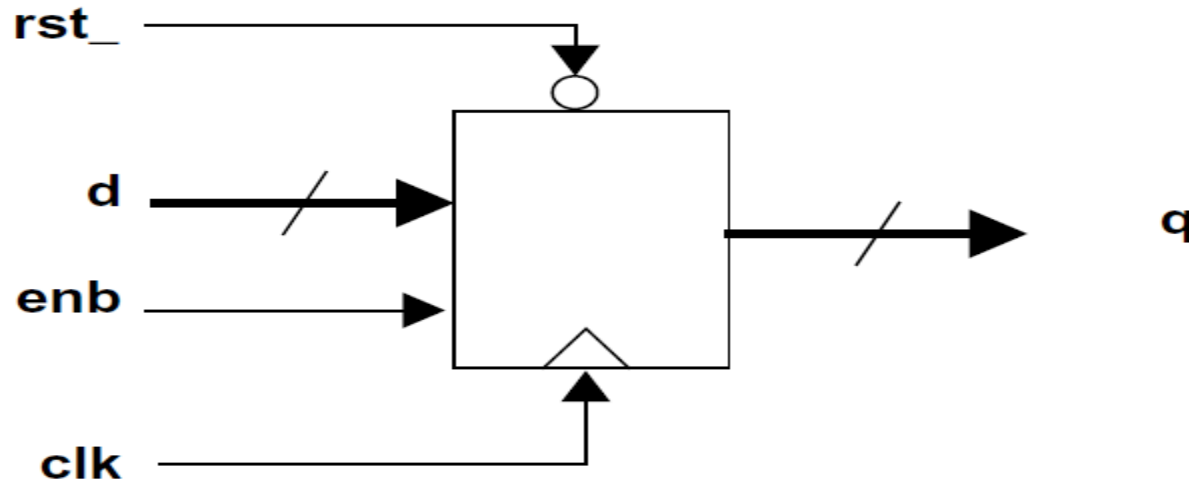
Design com SystemVerilog

Prof. Rômulo Calado Pantaleão Camara
CPU

Carga Horária: 2h/60h

Descrição Registrador

- ✓ Desenvolva um registrador



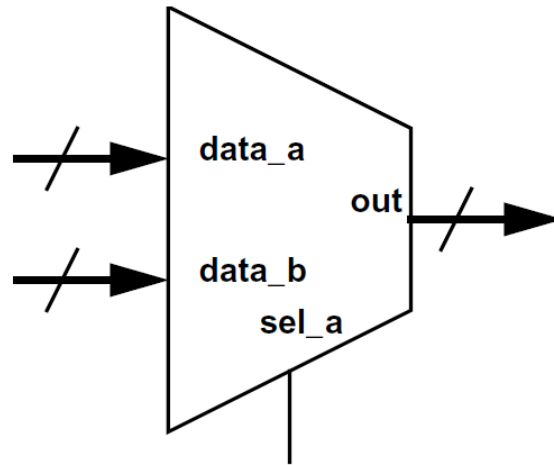
- ✓ *d* e *q* são vetores *logic*;
- ✓ *rst_* é assíncrono e ativo em *low*;
- ✓ O clock do registrador funciona na borda de subida:
 - Se *enb* estiver em alto, *d* é passado para *q*;
 - Caso contrário o registrador retêm o valor corrente;

Descrição Registrador

- ✓ Defina o módulo com o nome `register_m`;
- ✓ Parametrize a largura do barramento `WIDTH`;
- ✓ Utilize `timeunit` e `timeprecision`;
- ✓ Use `always_ff`;
- ✓ Opicional o uso do `iff`;
- ✓ Simule o registrador com o arquivo `testbench`;
- ✓ Verifique que passou em todos os testes;

Descrição MUX

- ✓ Desenvolva um multiplexador



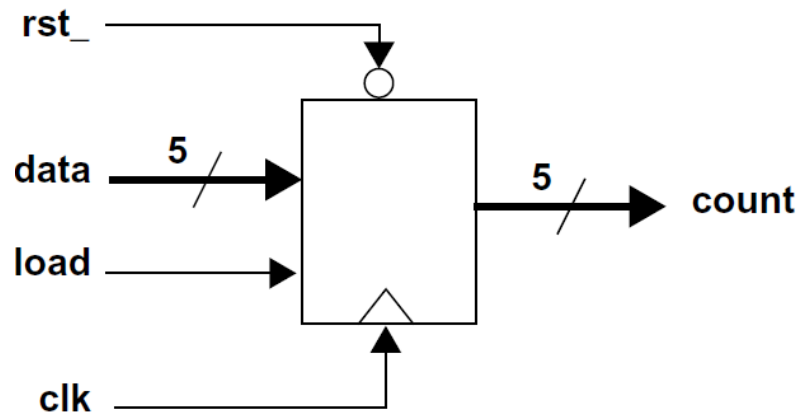
- ✓ Inputs e outputs são vetores *logic*;
- ✓ A entrada *sel_a* seleciona qual saída receberá out;
 - Se for 1, sai *data_a*;
 - Se for 0, sai *data_b*;

Descrição MUX

- ✓ Defina o módulo com o nome `mux_m`;
- ✓ Parametrize a largura do barramento `WIDTH`;
- ✓ Utilize `timeunit` e `timeprecision`;
- ✓ Use `always_comb`;
- ✓ Utilize `unique case` *ou* `unique if`;
- ✓ Use o default caso necessário;
- ✓ Simule o registrador com o arquivo `testbench`;
- ✓ Verifique que passou em todos os testes;

Descrição Contador

- ✓ Desenvolva um Contador



- ✓ Inputs e outputs são vetores *logic*;
- ✓ *rst_* é assíncrono e ativo em *low*;
- ✓ O clock do registrador funciona na borda de subida:
 - Se *load* estiver em alto, *data* é passado para *count*;
 - Caso contrário o contador é incrementado;

Descrição Contador

- ✓ Defina o módulo com o nome `counter_m`;
- ✓ Parametrize a largura do barramento `WIDTH` e ponha o valor *default* 1;
- ✓ Utilize `timeunit` e `timeprecision`;
- ✓ Use `always_comb`;
- ✓ Utilize `priority if`;
- ✓ Utilize `always_ff` para escrever um bloco síncrono;
- ✓ Simule o contador com o arquivo `testbench`;
- ✓ Verifique que passou em todos os testes;
- ✓ Será que pode ser utilizado o `(++)` ou `(+=)`?

Entrega do ULA

✓ Entregar até o dia 11/05/2014 às 23:59:59 os exercícios:

1. Registrador e seu testbench;
2. Mux e seu testbench;
3. Contador e seu testbench;
4. ULA Always_comb e seu testbench;
5. Symbol_to_chip e o testbench;

Todos os arquivos devem ser no formato .sv.