Instituto Tecnológico de Aeronáutica

Divisão de Ciência da Computação - IEC

Departamento de Sistemas de Computação - IEC-SC

EEA-25 - Sistemas Digitais Programáveis

Lista de Exercícios Nº 3

Prof. André da Fontoura Ponchet

São José dos Campos - SP

28 de agosto de 2023

Sumário

1 Exercícios Teóricos

1.1 Exercício 1 (1,0)

In a decoder, a logic expression is created for each output. Once all of the output logic expressions are found, how can the decoder logic be further minimized?

A) By using K-maps to find the output logic expressions.

B) By buffering the inputs so that they can drive a large number of other gates.

C) By identifying any logic terms that are used in multiple locations (inversions, product terms, and sum terms) and sharing the interim results among multiple circuits in the decoder.

D) By ignoring fan-out.

1.2 Exercício 2 (1,0)

If it is desired to have the outputs of an encoder produce 0's for all input codes not defined in the truth table, can "don't cares" be used when deriving the minimized logic expressions? Why?

A) No. Don't cares aren't used in encoders.

B) Yes. Don't cares can always be used in K-maps.

C) Yes. All that needs to be done is to treat each X as a 0 when forming the most minimal prime implicant.

D) No. Each cell in the K-map corresponding to an undefined input code needs to contain a 0 so don't cares are not applicable.

1.3 Exercício 3 (1,0)

What will always cause a digital storage device to come out of metastability and settle in one of its two stable states? Why?

A) The power supply. The power supply provides the necessary current for the device to overcome metastability.

B) Electrical noise. Noise will always push the storage device toward one state or another. Once the storage device starts moving toward one of its stable states, the positive feedback of the storage device will reinforce the transition until the output eventually comes to rest in a stable state.

C) A reset. A reset will put the device into a known stable state.

D) A rising edge of clock. The clock also puts the device into a known stable state.

1.4 Exercício 4 (1,0)

What was the purpose of replacing the inverters in the cross-coupled inverter pair with NOR gates to form the SR Latch?

A. NOR gates are easier to implement in CMOS.

B. To provide the additional output Qn.

C. To provide more drive strength for storing.

D. To provide inputs to explicitly set the value being stored.

1.5 Exercício 5 (1,0)

Identify all the errors on the following code:

```
// Find at least twelve things wrong!
module twelve_wrong (w, v, y, t, b, c, s)
input [3:0] a, b, c, s;
output [3,0] w, v, y, t;
reg [3:0] w, v, y, t,
assign W = s;
always @(a or b or c and s)
if (s == 0)
v = a;
y = b;
t = c;
else if (s <= 4'b0101)
begin
v = c;
y = b;
t = a;
a = s;
end
else if (s == 6 or s == 7)
v = c;
else v == 4'b'xxxx;
endmodule
```

2 Exercício de Simulação

2.1 Exercício 6 (5,0)

The VeriRISC CPU contains an accumulator register and an instruction register. One generic register definition can serve both purposes.

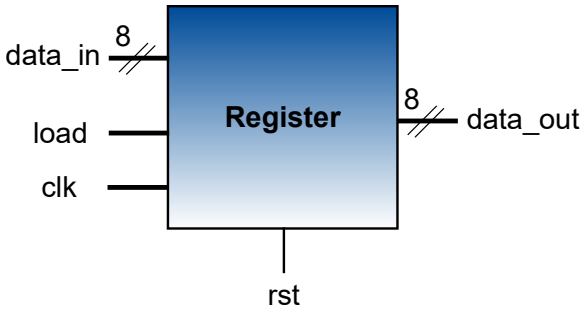

Figura 1: Register diagram.

Write the register model using the Verilog *always* procedural block. Specifications:

- *data_in* and *data_out* are 8-bit signals.
- *rst* is synchronous and active high.
- The register is clocked on the rising edge of clk.
- If load is high, the input data is passed to the output *data_out*.
- Otherwise, the current value of *data_out* is retained in the register.

Crie um arquivo com o nome *register.v* e verifique o seu funcionamento com o *testbench register_test.v*. Verifique o funcionamento do register através dos seguintes comandos no terminal:

$$\begin{cases} iverilog\ driver\_test.v \\ vvp\ a.out \\ gtkwave\ driver\_test.vcd \end{cases}$$

Apresente as capturas de tela do terminal e do *gtkwave*.