

Experimento 7

Máquina de estados síncrona Moore

OBJETIVOS:

- Projetar e implementar uma máquina de estados síncrona Moore em FPGA usando a linguagem VHDL.

PRÉ-RELATÓRIO:

- Apresente um diagrama de estados que descreva a máquina de estados proposta.
- Apresente o código VHDL de uma entidade e arquitetura que implementem, usando o modelo RTL, o diagrama de estados apresentado por você (ver exemplo de implementação de máquina de estados usando modelo RTL no documento tutorial fornecido pelo professor junto deste roteiro). Não serão corrigidos códigos VHDL que não estejam acompanhados de um diagrama de estados!
- Apresente o arquivo UCF para implementação dessa máquina de estados no FPGA.
- Lembre-se: seu pré-relatório deve ser entregue como um único arquivo PDF. Não anexe arquivos .vhd nem arquivos .ucf!

VISTO:

Implementar no FPGA uma máquina de estado síncrona do tipo Moore para controlar uma máquina de refrigerantes que aceita moedas de R\$ 0,25 e R\$ 0,50. A cada transição do *clock*, a máquina deve contar o dinheiro inserido e liberar o refrigerante (e o troco) assim que a soma totalizar ou exceder R\$ 1,00. A máquina deve aceitar qualquer combinação de moedas de R\$ 0,25 e R\$ 0,50, independentemente da ordem em que as moedas foram inseridas. A qualquer momento (desde que a contagem ainda não tenha alcançado R\$ 1,00) o usuário poderá cancelar a compra e a máquina deve, também na transição do *clock*, devolver a quantia inserida.

Considere que a máquina só dispõe de um sabor de refrigerante (ou que a escolha do refrigerante é feita antes da máquina de estados iniciar). Logo, o refrigerante é liberado automaticamente (mas na transição do *clock*) após a inserção do valor de R\$ 1,00 com ou sem troco, não sendo necessário pressionar nenhum botão após a inserção do montante para receber o refrigerante. Isto impede a possibilidade, por exemplo, da inserção do valor de R\$ 1,50.

A entidade VHDL deverá ter como entrada um vetor *A* de 2 bits — ligados a duas chaves quaisquer do kit de desenvolvimento (SW0 a SW7) — que indicará se foi inserida uma moeda de R\$ 0,25 (se $A = 01$), se foi inserida uma moeda de R\$ 0,50 (se $A = 10$), se foi solicitada o cancelamento da compra (se $A = 11$) ou se não houve nenhuma ação por parte do usuário (se $A = 00$).

Deverá ter como entrada também um *clock* de 1 bit — ligado a um botão qualquer do kit de desenvolvimento (BTN0 a BTN3) — que fará com que a máquina leia as entradas e mude (ou não) o estado e as saídas. A leitura das entradas, assim como a mudança de estado e das saídas, deverá acontecer exclusivamente na borda de subida do *clock*.

A entidade VHDL deverá ter três saídas de 1 bit cada — cada uma ligada a um LED diferente do kit de desenvolvimento (LD0 a LD7) — que indicarão, respectivamente, se a máquina liberou (ou não) um refrigerante, devolveu (ou não) uma moeda de R\$ 0,25 e(ou) devolveu (ou não) uma moeda de R\$ 0,50.

Na estrutura *process* que descreve a lógica dos registradores (memória da máquina de estados), inclua uma condição de “reset” (associada a uma entrada da entidade, ligada a um botão ou a uma chave do kit de desenvolvimento) que leve, de forma assíncrona (isto é, independentemente do sinal de *clock*) a máquina de estados de volta ao estado inicial. Isto facilitará o processo de teste no FPGA da máquina implementada. Um exemplo de

implementação de condição de “reset” é mostrada no documento tutorial fornecido pelo professor junto deste roteiro.

DICAS:

1. Antes de começar, certifique-se de que você sabe a diferença entre uma máquina de estados do tipo Moore e uma do tipo Mealy.
2. Em cada estado, haverá quatro possíveis combinações de entrada que devem ser consideradas: inseriu 25 centavos, inseriu 50 centavos, solicitou cancelamento da compra ou não fez nenhuma ação. Assim, para cada estado, haverá até quatro transições possíveis. Ao elaborar o diagrama de estados, lembre-se de considerar todas as quatro possibilidades em cada um dos vários estados!
3. Note que, após a liberação do refrigerante e(ou) devolução de moedas, nem sempre a máquina retornará ao estado inicial.
 - a. Imagine que, na transição do *clock*, o usuário solicitou o cancelamento da compra, levando a máquina ao estado que indica a devolução de R\$ 0,75; é possível que, na próxima transição do *clock*, o usuário já esteja inserindo uma moeda de R\$ 0,50. Neste caso, a máquina de estados não deve retornar ao estado inicial, mas sim ir direto ao estado que indica que a quantia inserida alcançou R\$ 0,50.
 - b. Entretanto, se a máquina liberou o refrigerante e(ou) devolveu moedas e, na próxima transição do *clock*, o usuário não inseriu moedas, ou então se ele solicitou um novo cancelamento de compra, a máquina deve sim retornar ao estado inicial. Pois, se a máquina permanecesse no mesmo estado, isso significaria liberar mais refrigerante (ou devolver mais moedas) indevidamente.
4. Seguindo estas especificações e dicas, é possível implementar a máquina com somente 9 estados.