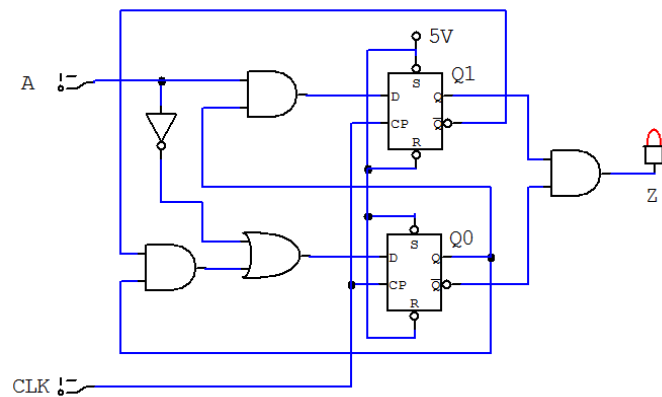


Sistemas Digitais 2 - 2/2015

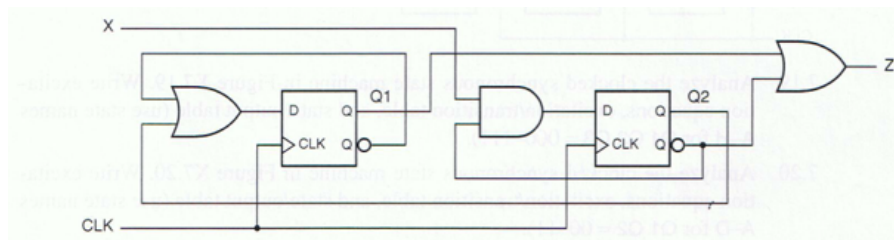
Lista de Exercícios – P1 Trezentos

1. O circuito mostrado na figura abaixo é de um detector de sequências. A partir da análise de máquina de estados (equações de excitação, tabela de transição e de saída e diagrama de estados):

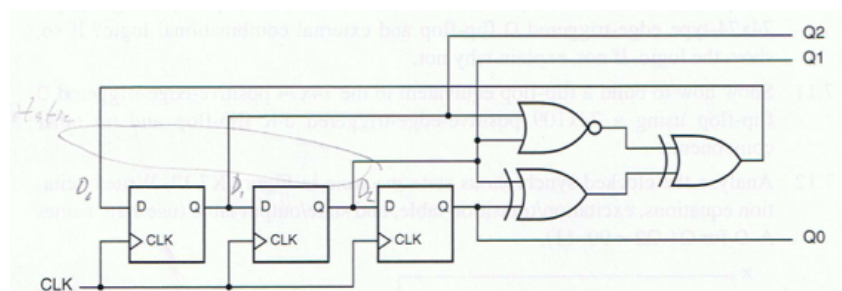
- Determine a sequência que o mesmo detecta;
- A máquina é de Mealy ou de Moore? Refaça o projeto usando o outro tipo.



2. Analise as máquinas de estados abaixo (equações de excitação, tabela de transição e de saída e diagrama de estados). Considere **Q2** como o bit mais significativo.



Fonte: Wakerly, John F., *Digital Design: Principles and Practices*, 4th ed., Prentice Hall, 2005.



Fonte: Wakerly, John F., *Digital Design: Principles and Practices*, 4th ed., Prentice Hall, 2005.

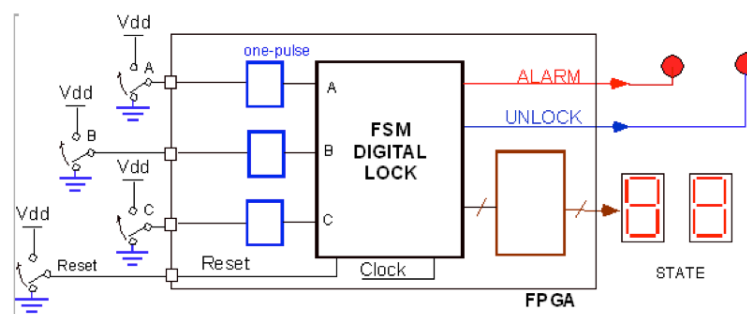
3. Você foi indicado para projetar uma máquina de vendas de refrigerante no seu centro acadêmico. Os refrigerantes são parcialmente financiados pelo IEEE, de maneira que cada um custa apenas **R\$ 0,25**. A máquina deve aceitar moedas de **R\$ 0,05, 0,10 e 0,25**. Quando a quantidade correta de moedas for colocada, a máquina deve entregar o refrigerante e devolver o troco, se necessário. Assumindo que apenas uma moeda é colocada em cada ciclo de *clock*, projete uma FSM para controlar a máquina de vendas. As entradas da máquina são **C, D e VC**, para indicar as moedas de **R\$ 0,05, 0,10 e 0,25**, respectivamente, e as saídas são **REFRI** (entrega

o refrigerante), **TROCO1C** (entrega uma moeda de 0,05 como troco), **TROCO1D** (entrega uma moeda de 0,10 como troco) e **TROCO2D** (entrega duas moedas de 0,10 como troco). Quando as entradas alcançam **R\$ 0,25**, o refrigerante é entregue e o troco é devolvido apropriadamente. Em seguida, a máquina fica pronta para receber outro pedido. **Esboce o diagrama e a tabela de estados da máquina projetada.**

4. Projete uma FSM com uma saída **Y** que será **1** quando a sequência **0,1,0,1** ocorrer na entrada **X**. Considere que o circuito detecta sequências sobrepostas.

- Faça o diagrama e a tabela de estados utilizando Moore;
- Faça o diagrama e a tabela de estados utilizando Mealy;
- Esboce as saídas **Z_{MOORE}** e **Z_{MEALY}** em um diagrama de tempo, considerando **X=0, X=1, X=0, X=1, X=0, X=1, X=0, X=1** em 8 ciclos de clock consecutivos e justifique sua resposta.

5. Suponha que você queira implementar em FPGA uma fechadura digital que possui 3 botões de entrada: **A, B e C**.



Assuma que os botões não podem ser pressionados simultaneamente, e que a fechadura possui as seguintes características:

- Quando é pressionada uma determinada combinação nas entradas, o sinal **UNLOCK** é acionado e a fechadura é aberta. Para encontrar a combinação da sua fechadura, use os 4 últimos dígitos do seu número de matrícula e considere a Tabela 1. Por exemplo, para um aluno com matrícula 09/0023432, a combinação será **A-B-A-C**.

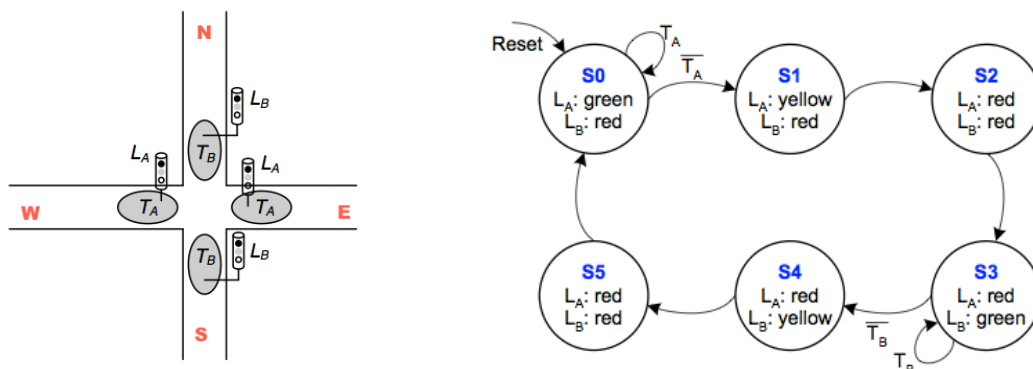
Tabela 1

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| A | B | C | A | B | C | A | B | C | A |

- Após aberta, a fechadura pode ser travada novamente pressionando qualquer botão. Considere que a combinação **A-A** funciona como *reset* a partir de qualquer estado, exceto quando o alarme tiver sido acionado. Caso A-A faça parte da sequência correta, utilize outra combinação, como **B-B** ou **C-C**. Para dificultar a descoberta da sequência correta, o alarme somente será acionado após a inserção de uma sequência incorreta completa (4 botões pressionados).
- A única maneira de desligar o alarme é pressionando a combinação **C-A**.

Esboce o diagrama e a tabela de estados, indicando o que cada estado representa, que condições são necessárias para a transição dos estados e quais são as saídas correspondentes. Enumere os estados como S0, S1, etc. Sinta-se à vontade para fazer suposições razoáveis que não foram abordadas na especificação, desde que as descreva na sua solução.

6. Considere o projeto do controlador de semáforos mostrado na figura abaixo, com período de clock igual a 5 segundos.

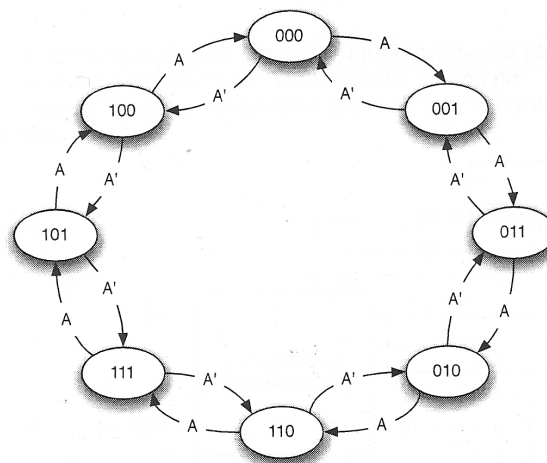


Altere o projeto de maneira a incluir quatro novas entradas: P_A e P_B , que indicam que há um pedestre tentando atravessar a pista **EW** e **NS**, respectivamente, e **TMS** e **TML**, que indicam que **5** e **30 segundos** se passaram, respectivamente. Além disso, o clock do novo projeto deve ser de **1 Hz**. Quando uma determinada pista estiver liberada (por exemplo, no caso da pista **EW**, $L_A = \text{green}$ e $L_B = \text{red}$), o sentido de tráfego deve ser obrigatoriamente alterado em 3 casos:

1. Se não houver nenhum carro trafegando nesta pista ($T_A = 0$) e já passaram **30 segundos** ($TML = 1$);
2. Se, passado o tempo mínimo de **5 segundos** ($TMS = 1$), não houver nenhum carro trafegando nesta pista ($T_A = 0$) e houver carros esperando na outra pista ($T_B = 1$);
3. Se, passado o tempo mínimo de **5 segundos** ($TMS = 1$), houver algum pedestre tentando atravessar esta pista ($P_A = 1$).

Portanto, cada pista deve ficar liberada por no mínimo **5 segundos**. Se após **30 segundos** ainda há carros trafegando em uma pista que está liberada ($T_A = 1$) e não há nenhum carro esperando na outra pista ($T_B = 0$) e nem pedestres tentando atravessar ($P_A = 0$), então ela continua liberada. Ao se alternar o sentido de tráfego, o semáforo deve ficar **2 segundos** no amarelo e a seguir ambas as pistas devem ficar fechadas por **1 segundo**, para que finalmente o tráfego seja liberado no outro sentido. Considere também que os contadores que geram os sinais **TMS** e **TML** são zerados no instante em que ambos os semáforos estão fechados. **O novo projeto deve conter a tabela e o diagrama de estados. Use máquina de Moore.**

7. A figura abaixo mostra o diagrama de uma máquina de estados que possui uma entrada **A**. No diagrama, A' é o complementar de **A**. Utilizando flip-flops do tipo D, implemente essa máquina de estados e, com base no diagrama de estados, explique o que ela faz.



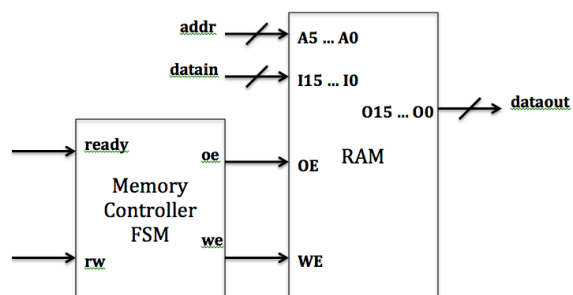
8. Usando multiplexadores e flip-flops tipo D, esboce o esquemático de um registrador de 4 bits com carga paralela e deslocamento à direita e à esquerda de acordo com a seguinte tabela:

| S1 | S0 | Operação |
|----|----|-------------|
| 0 | 0 | Hold |
| 0 | 1 | Shift Right |
| 1 | 0 | Shift Left |
| 1 | 1 | Load |

9. Esboce o esquemático de um registrador de 4 bits, usando multiplexadores e FF's D, que funciona conforme mostra a tabela abaixo:

| S1 | S0 | Operação |
|----|----|---|
| 0 | 0 | Hold |
| 0 | 1 | Desloca dois bits à direita (entrando 0's) |
| 1 | 0 | Desloca dois bits à esquerda (entrando 1's) |
| 1 | 1 | Load |

10. A figura abaixo mostra um controlador de RAM que deve ter 4 estados: **espera**, **decisao**, **leitura** e **escrita**. Inicialmente, o controlador está no estado **espera** aguardando a indicação de que a memória está pronta para receber um comando (escrever ou ler). Caso esteja, o controlador vai para o estado **decisao**, de onde pode ir para o estado **leitura** ou **escrita**, dependendo do valor do sinal **rw**. Nos estados **leitura** e **escrita** são realizadas as operações correspondentes.



ready: indica que a memória está pronta para efetuar uma operação (**ready = 1**)
rw: requisita uma operação de leitura (**rw = 1**) ou escrita (**rw = 0**)
we: controla a entrada **WE** da RAM
oe: controla a entrada **OE** da RAM
WE: habilita escrita (**WE = 1**) ou leitura (**WE = 0**) na RAM
OE: habilita (**OE = 1**) ou desabilita (**OE = 0**) a saída da RAM

- Qual é a capacidade da RAM em palavras, bytes e bits? Justifique;
- Esboce o diagrama e a tabela de estados do controlador usando Moore.