

**EEL5105 - Circuitos e Técnicas Digitais**      Prof. Fabian L. Cabrera  
Exercícios – Aula 8: Máquinas de estados

Nos seguintes exercícios\*: (a) Desenhe o diagrama de estados. (b) Escreva a codificação dos estados. (c) Faça a tabela de transição e otimize a lógica combinacional\*\*. (d) Implemente o circuito usando flip-flops tipo D e portas lógicas.

\* Todos os circuitos nos exercícios dispõem de um sinal de **clock** e de **reset**, implicitamente.

\*\* Use mapas de Karnaugh quando houver 3 ou mais variáveis no lado esquerdo da tabela de transição. Use simplificação algébrica em caso contrário.

1. **Verificador de bit de paridade:** Projete um circuito que recebe um número serialmente, ou seja, somente um bit **b** é recebido cada ciclo de relógio a começar pelo bit mais significativo (MSB). O circuito tem uma saída **y**, a qual adota o valor de '0' quando o número de '1's recebido até o momento é par. Por outro lado, a saída **y** adota o valor de '1' quando o número de '1's recebido até o momento é ímpar.

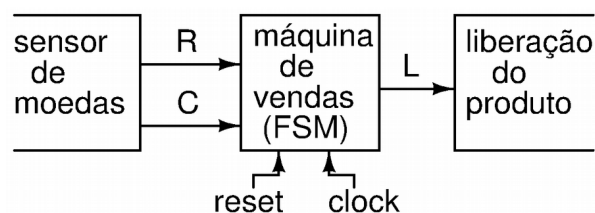
2. **Somador serial:** Projete um circuito que recebe dois números A e B serialmente, ou seja, somente são recebidos dois bits **a** e **b**, respectivamente, a cada ciclo de relógio a começar pelos bits menos significativos (LSB). O circuito tem uma saída **s**, a qual deve informar o bit de soma correspondente à posição dos bits recebidos **a** e **b**. Por tanto,  $s = a + b$  quando não há carry do estágio anterior.  $s = a + b + '1'$  quando há carry do estágio anterior.

3. **Contador 1:** Projete um contador que possui uma saída (W de 2 bits) e 3 estados: M, Y e Z. O estado do contador após o reset é Y. A cada borda de subida do relógio, o contador avança de M para Y, de Y para Z e de Z para M. No estado M, W é "00". No estado Y, W é "10". No estado Z, W é "11".

4. **Contador 2:** Projete um contador que possui uma saída (W de 2 bits) e 3 estados: M, Y e Z. O estado do contador após o reset é M. O circuito possui uma entrada **k**, se  $k = '0'$  o contador avança de M para Y, de Y para Z e de Z para M. Por outro lado, se  $k = '1'$  o contador avança de Y para M, de M para Z e de Z para Y. No estado M, W é "00". No estado Y, W é "10". No estado Z, W é "01".

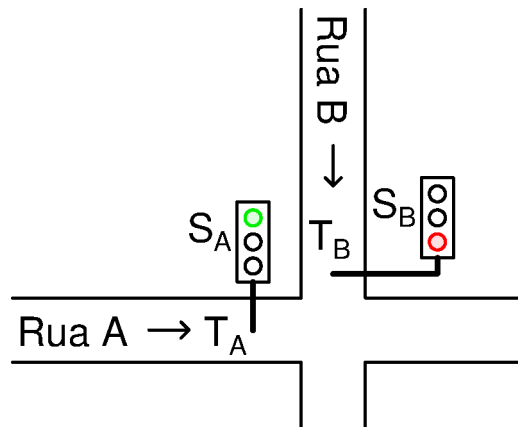
5. **Contador 3:** Projete um contador com uma saída (P de 3 bits) que segue a sequência: "010", "001", "011", "100", "110". No final da sequência, a saída volta ao primeiro valor repetendo indefinidamente. No reset, a saída começa no valor "010".

6. **Máquina de vendas:** Projete uma máquina que libera um produto ( $L = '1'$ ) quando o usuário deposita o seu preço em moedas, o preço do produto é R\$1,50. Apenas moedas de R\$1 e de R\$0,5 pode ser inseridas, uma por vez. Quando uma moeda de R\$1 é inserida o sinal R é ativado ( $R = '1'$ ). Quando uma moeda de R\$0,50 é inserida o sinal C é ativado ( $C = '1'$ ). Caso o valor inserido supere R\$1,50, a máquina entrega o produto mas não devolve o troco.



7. **Semáforo:** No cruzamento entre duas ruas A e B há um par de semáforos controlados pelos sinais digitais  $S_A$  e  $S_B$  (cad um de 2 bits). Também foram instalados em cada rua, sensores de tráfego ( $T_A$  e  $T_B$ ), sendo que  $T_A=1$  quando há tráfego na rua A e  $T_B=1$  quando há tráfego na rua B. O comportamento dos semáforos deve ser o seguinte:

- Se  $S_A$  está em “verde”, ele continua assim enquanto houver tráfego na rua A.
- Se  $S_B$  está em “verde”, ele continua assim enquanto houver tráfego na rua B.
- A forma como os semáforos mudam de cor é a seguinte:  
 (Verde, Vermelho) →  
 (Amarelo, Vermelho) →  
 (Vermelho, Verde)



8. **Comparador serial:** Projete um circuito que recebe dois números A e B serialmente, ou seja, somente são recebidos dois bits **a** e **b**, respectivamente, a cada ciclo de relógio a começar pelos bits mais significativos (MSB). O circuito tem uma saída (**c** de 2 bits). O valor de **c** deve ser:

- 00 se os dois números são iguais
- 10 se A é maior do que B
- 01 se B é maior do que A

9. **Detector de sequência:** Projete um detector de sequência que recebe um bit de entrada serial **m** (um bit a cada ciclo de relógio, começando pelo bit mais significativo - MSB) e gera uma saída T de 1 bit.  $T=1$  se as entradas recebidas seguirem a 01011, caso contrário  $T=0$ . Use nomes com letras para cada estado, por exemplo: A, B, C ...