PMR3303 – Eletrônica Digital para Mecatrônica

Prof. Dr. Celso M. Furukawa – Lista 13 (rev. A) – Entrega até 27/05/2019

Os exercícios E1 a E5 a seguir se referem às máquinas de estados dos controladores descritos na apostila da experiência 6. Sugerimos que você transcreva os projetos para Verilog e os simule no EDAplayground.

E1. A partir do diagrama de estados do controlador xxTmrUC do temporizador, comprove que as expressões para os bits de saída En, Ld e T válidas em cada estado são:
WAIT: En = St, Ld = St T = 0.
LOAD: En = 1, Ld = 0 T = 1.
RUN: En = 1, Ld = 0 T = ~RC

E2. Projete o circuito do controlador xxTmrUC do temporizador usando flip-flops D. Codifique os estados da seguinte forma: WAIT=00, LOAD=01 e RUN=11. O estado não usado 10 deve zerar todas as saídas e levar o controlador para o estado WAIT. Determine as equações simplificadas das saídas e de excitação dos flip-flops. Desenhe o diagrama lógico.

E3. Modifique o controlador das bombas (xxBombas) para que, quando o nível cair abaixo de LL, a bomba auxiliar BA seja ligada e permaneça assim até que o nível fique acima de H. Adote a mesma codificação dos estados usada na apostila. Use um flip-flop JK para implementar o bit Q_1 e um flip-flop T para Q_0. Determine as equações simplificadas excitação dos flip-flops (Veja o exemplo no eDisciplinas).

E4. Reprojete o controlador das bombas (xxBombas) para que detecte variações improváveis de nível (exemplo: N_1N_0=00 passando para 11), indo para um estado de falha *SF* em que desliga as bombas e ativa uma saída adicional F e ficando nesse estado até ser resetado. Serão necessários mais estados adicionais. Por exemplo, o estado SB deve ser desmembrado em dois, SB0 e SB1, como mostra a figura f4. O controlador vai para SB0 quando o nível cai de 00 para 10 e deve passar para SB1 se o nível retornar a 00, ou passa para outro estado se o nível cair para 11. Se ocorrer uma leitura de nível 01 no estado em SB0, considera-se que houve uma falha. Será necessário também um estado inicial SI que faça a primeira leitura de nível e passe para um estado adequado; no exemplo da figura, o controlador vai para o estado SB0 se a primeira leitura for 10. Faça o diagrama de estados do novo controlador.

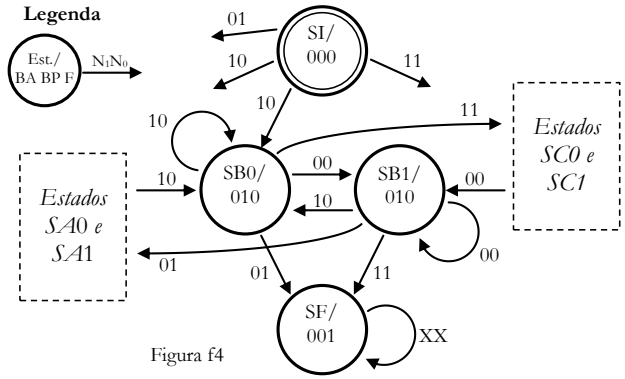
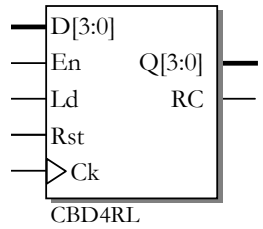


Figura f4

E5. Mostre como implementar o contador regressivo com carga em paralelo e reset síncrono CBD4RL mostrado na figura f5, usando flip-flops D e circuitos lógicos adicionais.



CBD4RL

Rst	En	Ld	Ck	$Q[3:0]^{k+1}$	RC	
1	X	X	↑	1111	0	(reset síncrono)
0	0	X	X	$Q[3:0]^k$	0	(contador desabilitado)
0	1	0	↑	$Q[3:0]^k - 1$	Q==0000	(decremento)
0	1	1	↑	D[3:0]	Q==0000	(carga em paralelo)

Figura f5

E6. A figura f6 mostra um registrador de 4 bits, onde a entrada CE (*Clock Enable*) habilita a carga em paralelo quando em 1. Mostre como implementar o registrador usando flip-flops tipo D comuns (sem entrada CE para habilitar/desabilitar o *clock*) e portas lógicas adicionais.

E7. O registrador descrito pela figura f7 tem capacidade de incremente e deslocamento, conforme mostra a tabela. Mostre como construir esse registrador usando o registrador do exercício anterior (figura f6) e circuitos lógicos adicionais.

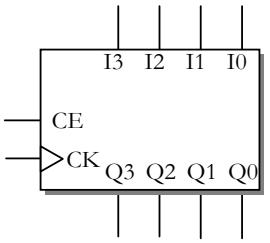


Figura f6

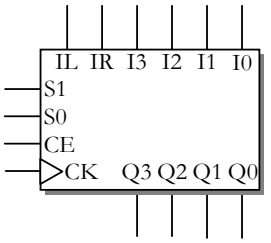


Figura f7

CE	S1S0	CK	Q3	Q2	Q1	Q0
0	XX	X	Q3	Q2	Q1	Q0
1	00	↑	I3	I2	I1	I0
1	01	↑	(Q3Q2Q1Q0) + 1			
1	10	↑	IL	Q3	Q2	Q1
1	11	↑	Q2	Q1	Q0	IR

E8. Usando o registrador da figura E6 e portas lógicas adicionais, construa um **registrador divisor inteiro** com uma entrada DV_LD tal que quando DV_LD = 0 o registrador é carregado em paralelo com o valor das entradas I[3:0]; quando DV_LD = 1 o registrador divide por 2 o valor armazenado **em complemento de 2** (o resto da divisão é descartado)

Algumas respostas

E2) Saídas: En = Q0 + Q1'.St, Ld = Q1'.Q0'.St, T = Q1'.Q0 + Q0.RC'.
Flip-flops: D0 = Q0.RC' + Q1'.Q0 + Q1'.St, D1 = Q0.RC' + Q1'.Q0.
E3) J = N1.N0.Q0, K = Q0' + N1'.N0; T0 = N1.Q1'.Q0' + N1'.N0.Q0
E4) Transições (Qk→[Qk+1 para N1N0=00, 01, 11, 10]): SI→[SA0, SA1, SC0, SB0];
SA0→[SA0, SA1, SF,SB0]; SA1→[SA0, SA1, SF, SF]; SC0→[SF, SF, SC0, SC1]; SC0→[SB1, SF, SC0, SC1].
Saídas em SA0 e SA1: 000; em SC0 e SC1: 110.
E5) Entrada D do ff de Q_i: Rst'.(En'.Q_i + En.Ld.D_i + En.Ld'.$Q_i \oplus Q_{i-1}$'.Q_{i-2}'...Q_0')
Nota: o termo entre parêntesis é equivalente a um multiplexador 4:1 com $\{S_1, S_0\}$ = {En, Ld} e $I_0=I_1=Q_i$, $I_3=D_i$ e $I_2=Q_i \oplus Q_{i-1}$'...Q_0'
E6) Di =CE'.Qi + CE.Di (equivalente a um mux 2:1)
E7) Usando mux 4:1 em cada entrada I_i; entradas do mux [0, 1, 2, 3] = [I_i, $Q_i \oplus (Q_{i-1}.Q_{i-2}...Q_0)$, Q_{i+1}, Q_{i-1}]; com $Q_4 \equiv$ IL e $Q_{-1} \equiv$ IR.
E8) Se DV_LD == 1, $I_3 \leftarrow Q_3$ e demais $I_i \leftarrow Q_{i+1}$; senão $I_i \leftarrow I_i$ externo