

Universidade Federal de Santa Catarina
EEL51015: Circuitos e Técnicas Digitais
Semestre: 2023-2 Exercício 3

Circuitos Sequenciais: Projeto FSM

Tarefa: Implementar um circuito que mostre a sequência **cíclica** de 8 estados em **hexadecimal**. Cada aluno/dupla deve fazer a sequência da lista abaixo que será indicada pelo professor. Para obter dito circuito use a abordagem de resolução de projeto FSM aprendida na teoria. O sistema deve possuir uma entrada de clock e um reset em $KEY(0)$. A sequência deve ser mostrada numa saída S de 4 bits em formato binário nos $LEDR(3 \dots 0)$ e em formato hexadecimal no $Display HEX0$.

- Ao final da aula, entregar os seguintes passos para resolução do projeto FSM:
 - (i) Diagrama de estados [0,5]; (ii) Codificação de estados [0,5]; (iii) Tabela de transição de estados [1,0]; (iv) Mapas para obter circuito lógico do próximo estado e saídas [1,0]; (v) Desenho do Circuito [1,0].

Para orientação do aluno: **Há um exemplo parcialmente resolvido na página seguinte.**

- Implementar o circuito no Quartus usando VHDL [4,0 pontos]
 - Sugestões: utilizar o modelo de topo.vhd da tarefa do laboratório 5, modelo de decodificador *decod7seg* obtido no laboratório 4, e modelo de registrador de 4 bits obtido laboratório 5;
 - Enviar códigos VHDL pelo Moodle para posterior averigação.
- Verificar e mostrar ao professor o funcionamento do circuito lógico-sequencial na FPGA ($DE1$ ou $DE2$). Na placa: o botão **KEY(1)** deve ser usado como **sinal de relógio** (clock). [2,0 pontos]
 - A simulação via ModelSim ou emulador é opcional: porém se aluno que optar por mostrar o funcionamento **apenas** no emulador ou no ModelSim receberá **apenas 40% da pontuação total desse item**. Nesse caso pode realizar simulação com entrada de relógio CLK_1Hz ;

Sequências:

A – $0 \rightarrow 9 \rightarrow 4 \rightarrow b \rightarrow C \rightarrow 6 \rightarrow 1 \rightarrow F \rightarrow 0 \dots$
B – $0 \rightarrow b \rightarrow A \rightarrow 2 \rightarrow 3 \rightarrow C \rightarrow 7 \rightarrow F \rightarrow 0 \dots$
C – $0 \rightarrow 1 \rightarrow b \rightarrow 3 \rightarrow 4 \rightarrow d \rightarrow 8 \rightarrow F \rightarrow 0 \dots$
D – $0 \rightarrow 2 \rightarrow C \rightarrow 4 \rightarrow 5 \rightarrow E \rightarrow 9 \rightarrow F \rightarrow 0 \dots$
E – $0 \rightarrow 3 \rightarrow d \rightarrow 5 \rightarrow 6 \rightarrow b \rightarrow A \rightarrow F \rightarrow 0 \dots$
F – $0 \rightarrow 4 \rightarrow E \rightarrow 6 \rightarrow 7 \rightarrow 1 \rightarrow b \rightarrow F \rightarrow 0 \dots$
G – $0 \rightarrow 5 \rightarrow 3 \rightarrow 7 \rightarrow 8 \rightarrow 2 \rightarrow C \rightarrow F \rightarrow 0 \dots$
H – $0 \rightarrow 6 \rightarrow 1 \rightarrow 8 \rightarrow 9 \rightarrow 3 \rightarrow d \rightarrow F \rightarrow 0 \dots$
I – $0 \rightarrow 7 \rightarrow 2 \rightarrow 9 \rightarrow A \rightarrow 4 \rightarrow E \rightarrow F \rightarrow 0 \dots$
J – $0 \rightarrow 8 \rightarrow 3 \rightarrow A \rightarrow b \rightarrow 5 \rightarrow C \rightarrow F \rightarrow 0 \dots$

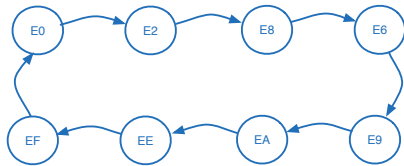
Dica: Exemplo parcialmente solucionado da sequência $0 \rightarrow 2 \rightarrow 8 \rightarrow 6 \rightarrow 9 \rightarrow A \rightarrow E \rightarrow F \rightarrow 0 \dots$ para orientação do aluno.

(FSM Exemplo)

Diagrama de estados

Codificação estados

Tabela transição estados



EA	s3	s2	s1	s0
E0	0	0	0	0
E2	0	0	1	0
E6	0	1	1	0
E8	1	0	0	0
E9	1	0	0	1
EA	1	0	1	0
EE	1	1	1	0
EF	1	1	1	1

Entradas				Saídas			
s3	s2	s1	s0	n3	n2	n1	n0
0	0	0	0	0	0	1	0
0	0	1	0	1	0	0	0
1	0	0	0	0	1	1	0
0	1	1	0	1	0	0	1
1	0	0	1	1	0	1	0
1	0	1	0	1	1	1	0
1	1	1	0	1	1	1	1
1	1	1	1	0	0	0	0

Estado atual (EA)=s3s2s1s0

Proximo Estado (PE)=n3n2n1n0

O estado atual será a saída do sistema

Obtenção da lógica do próximo estado

Desenho do circuito

VHDL do bit n(3) direto no usertop.vhd

n3	s3s2	00	01	11	10
	00	0	—	—	1
	01	—	—	—	1
	11	—	—	0	1
	10	0	1	—	1

$n3 = s1s0' + s1's0 = s1 \oplus s0$

n2	s3s2	00	01	11	10
	00	0	—	—	0
	01	—	—	—	0
	11	—	—	0	1
	10	1	0	—	1

$n2 = \dots$

n1	s3s2	00	01	11	10
	00	1	—	—	0
	01	—	—	—	0
	11	—	—	0	1
	10	1	1	—	1

n0	s3s2	00	01	11	10
	00	0	—	—	0
	01	—	—	—	1
	11	—	—	0	1
	10	0	0	—	0

