

**UNIVERSIDADE FEDERAL DA FRONTEIRA SUL  
CAMPUS CHAPECÓ  
CURSO DE CIÊNCIA DA COMPUTAÇÃO**

**ANDREI CARLESSO CAMILOTTO**

**TRABALHO MÁQUINAS DE ESTADO  
SISTEMAS DIGITAIS**

**CHAPECÓ  
2023**

## **Sumário**

<b>1 INTRODUÇÃO.....</b>	<b>2</b>
1.1 PROBLEMAS APRESENTADOS.....	2
<b>2 DESENVOLVIMENTO.....</b>	<b>3</b>
2.1 DESCRIÇÃO DO CIRCUITO.....	3
2.2 DIAGRAMA DA MÁQUINA DE ESTADOS.....	3
2.3 TABELAS DE ESTADOS.....	4
2.4 CIRCUITO FINAL.....	6
<b>3 CONSIDERAÇÕES FINAIS.....</b>	<b>7</b>
<b>REFERÊNCIAS.....</b>	<b>8</b>

## 1 INTRODUÇÃO

Seguindo as orientações do Professor Geomar Andre Schreiner foi realizado o trabalho referente à implementação de máquinas de estado no dia 28 e 29 de junho cujo objetivo é a solução de um dos três problemas apresentados na descrição da atividade, sendo que, para tal, é necessário a construção de um diagrama de máquina de estados, modelo Moore ou Mealy, extrair as tabelas de transição de estados e apresentar o funcionamento de uma implementação em linguagem de descrição de hardware ao professor.

### 1.1 PROBLEMAS APRESENTADOS

Dos três problemas propostos para atividade, optei por realizar o primeiro, o qual descreve o seguinte:

“Um sistema computacional possui um barramento ao qual estão conectados 4 dispositivos de entrada e saída, doravante denominados disp0, disp1, disp2 e disp3. O acesso destes dispositivos ao barramento obedece a uma prioridade, de modo que disp0 possui a mais alta prioridade, enquanto disp3 possui a mais baixa. Cada dispositivo possui um sinal de saída, “req”, e um sinal de entrada, “aut”. (No caso de disp0, “req0” e “aut0”). Quando o dispositivo necessita acessar o barramento ele faz seu sinal “req” valer “1”. Se o árbitro atender à requisição, ele levantará o sinal “aut” do dispositivo, mantendo-o em “1” até que o dispositivo libere o barramento. Para liberar o barramento, o dispositivo deve baixar seu sinal “req”. Considere que o sinal “req” somente é atendido na borda ativa do relógio. Considere também que o sinal “aut” muda de valor somente na (após a) borda ativa do relógio. Projete este árbitro de barramento.”

## 2 DESENVOLVIMENTO

### 2.1 DESCRIÇÃO DO CIRCUITO

O árbitro deverá conter:

- Entradas: rec0, rec1 e rec2, as quais recebem os requerimento dos dispositivos 0, 1 e 2, respectivamente.
- Saídas: aut0, aut1 e aut2, as quais enviarão um sinal ao dispositivo que tiver seu requerimento deferido.

Ele deverá ser responsável por deferir ou indeferir a requisição de uso do barramento pelo dispositivo, seguindo a ordem de preferência: Dispositivo 0 possui prioridade sobre o dispositivo 1 que possui prioridade sobre o dispositivo 2.

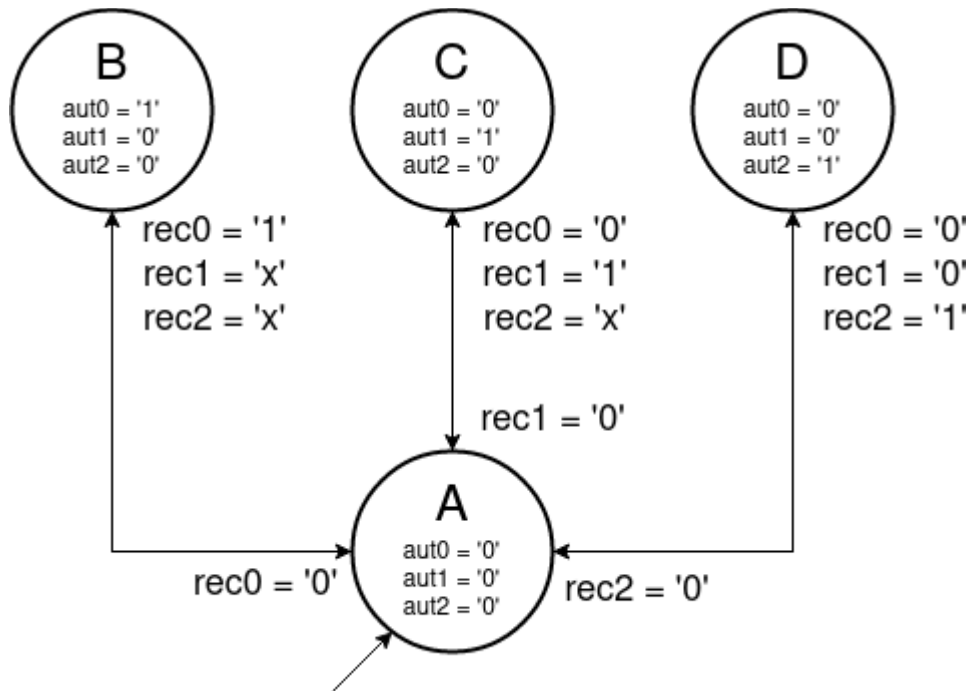
O árbitro receberá as solicitações dos dispositivos simultaneamente e, ao sinal alto de um clock, deve deferir o uso do barramento ao dispositivo de maior prioridade, o qual continuará em uso do dispositivo selecionado até ele desative seu sinal de requerimento, e, ao sinal do próximo clock, o árbitro irá retirar a permissão do dispositivo, e aguardará o sinal de clock seguinte para reatribuir a permissão a qualquer que seja o dispositivo de maior prioridade que requisitar o uso do barramento.

Note que em nenhuma circunstância o árbitro pode autorizar dois dispositivos simultaneamente, nem remover a autorização de uso já deferida a um dispositivo sem que ele desligue o sinal de requerimento.

### 2.2 DIAGRAMA DA MÁQUINA DE ESTADOS

A máquina de estados proposta segue o modelo proposto por Moore, onde as saídas do circuito dependem exclusivamente do estado atual. Para o árbitro de barramentos foi projetada uma máquina com quatro estados: A, B, C, D e E. Onde:

O estado “A” representa o estado inicial do circuito, onde nenhuma permissão é deferida. O estado “B”, “C” e “D” representam o deferimento do uso do barramento ao dispositivo 0, 1 e 2, respectivamente.



*Figura 1: Diagrama de Moore para o árbitro de barramento.*

Note que o árbitro deverá utilizar o estado inicial “A” como intermediário entre os outros três estados, portanto, utilizará dois ciclos de clock para remover a autorização de um dispositivo e deferir a autorização para outro.

Por exemplo: Suponha que o estado atual seja “B” e o dispositivo 0 esteja utilizando o barramento, para que o estado seja transitado para “C” é necessário que o dispositivo 0 encerre seu sinal de requerimento (req0) fazendo que o árbitro retorne ao estado “A” no ciclo de clock, e ao sinal do clock seguinte o dispositivo 1 deverá estar com o sinal de requerimento (req1) ativo para que o estado seja transitado para C. Totalizando, no mínimo, dois ciclos de clock para realizar a transferência.

### 2.3 TABELAS DE ESTADOS

A tabela que indica quais saídas cada estado possui é a seguinte:

*Tabela 1: Saídas dos estados*

Estado Atual	Saídas
A	aut0 = '0'

	aut1 = '0'
	aut2 = '0'
	aut0 = '1'
B	aut1 = '0'
	aut2 = '0'
	aut0 = '0'
C	aut1 = '1'
	aut2 = '0'
	aut0 = '0'
D	aut1 = '0'
	aut2 = '1'

---

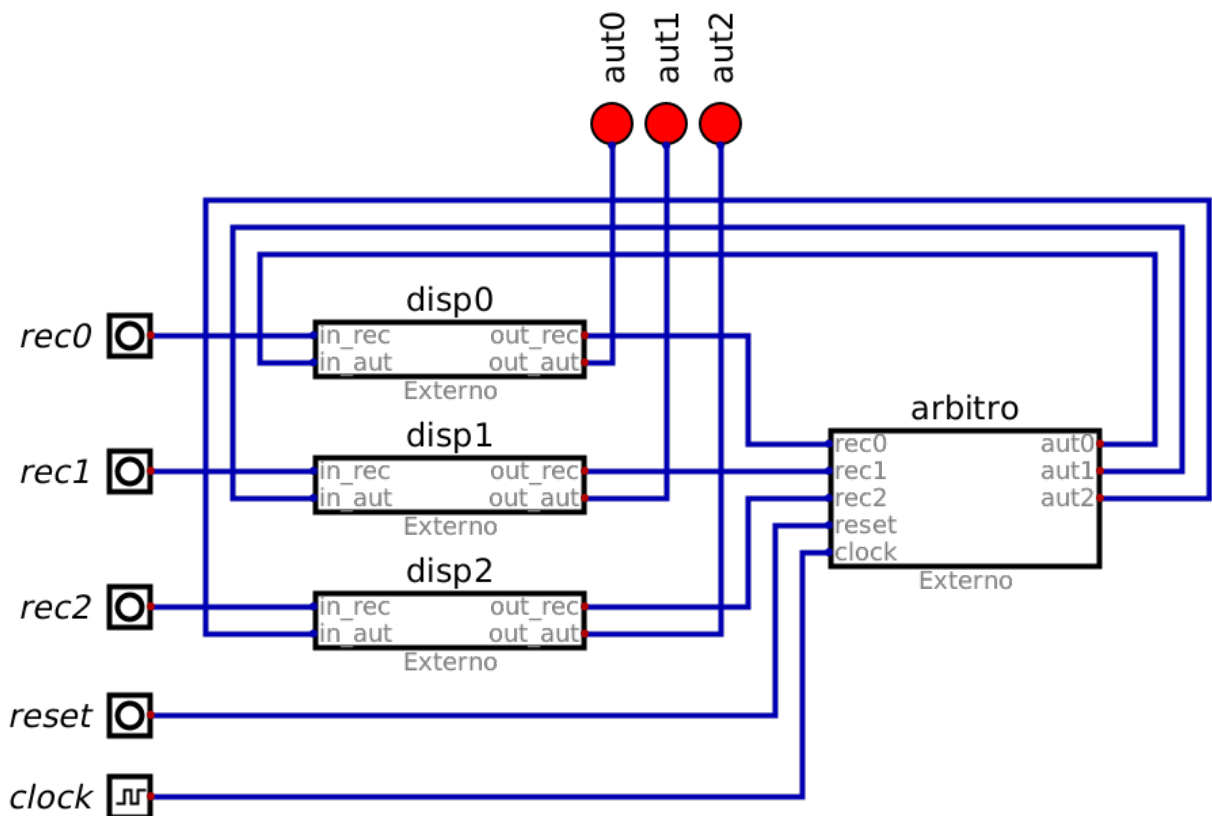
A tabela que indica quais valores de entrada levam aos respectivos estados é a seguinte:

*Tabela 2: Lógica de transição de estados*

Estado Atual	Lógica de transição	Próximo estado
A	req0 = '1'	B
	req1 = 'x'	
	req2 = 'x'	
B	req0 = '0'	A
	req1 = 'x'	
	req2 = 'x'	
A	req0 = '0'	C
	req1 = '1'	
	req2 = 'x'	
C	req0 = 'x'	A
	req1 = '0'	
	req2 = 'x'	
A	req0 = '0'	D
	req1 = '0'	
	req2 = '1'	
D	req0 = 'x'	A
	req1 = 'x'	
	req2 = '0'	

## 2.4 CIRCUITO FINAL

Fazendo o uso do diagrama de estados e das tabelas de transição foi desenvolvido a implementação em VHDL do árbitro e fazendo o uso do Software “Digital” a implementação foi testada e a imagem do circuito final foi essa:



*Figura 2: Circuito do árbitro.*

Tanto a implementação em VHDL, quanto a implementação no Digital estão disponíveis para consulta em anexo a este documento.

### **3 CONSIDERAÇÕES FINAIS**

Portanto, com a realização da máquina de estados, extração das tabelas e testes realizados com a implementação VHDL. Constata-se que os requerimentos descritos na sessão 1.1 foram concluídos. O árbitro responsável por deferir ou indeferir as requisições de uso de um barramento por dispositivos com ordem de prioridade foi implementado com sucesso, portanto, concluo este trabalho sem mais nada a comentar.



## REFERÊNCIAS

CAIMI, Luciano L. Slides. Slides – Sistemas Digitais, Moodle UFFS.