



ARQUITETURA DE COMPUTADORES | 21010 | 2023/2024

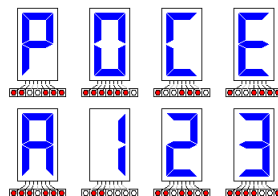
Pretendemos uma implementação em Digital Works de um sistema digital com o jogo da Memória, mas com regras especiais. O jogo vai sendo construído em 4 alíneas, valendo 1 valor cada alínea.

Alínea A (1 valor)

Fazer um componente para decodificar o código de uma carta e mostrar a carta em dois displays de 7 segmentos. Uma carta tem um naipe e um número. Considere que existem:

- 4 naipes (Paus, Ouros, Copas, Espadas)
- 4 números (Às, 2, 3, 4)

Um naipe e um número utilizam um código sequencial de 0 a 3, pela ordem listada, e são mostrados num display de 7 segmentos, da seguinte forma:



Circuito combinatório:

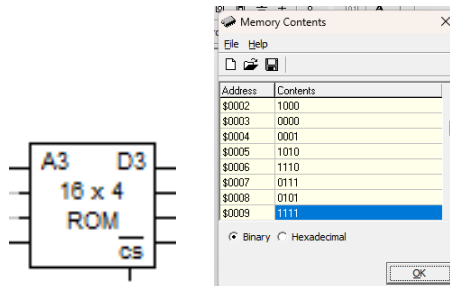
- Entradas:
 - Naipe: 2 bits - naipe codificado
 - Número: 2 bits - número codificado
 - EN: 1 bit - sinal de enable (não mostrar nada se inativo)
- Saídas:
 - Número: 7 bits - número decodificado
 - Naipe: 7 bits - naipe decodificado

Atendendo a que as saídas se destinam a ligar a displays de 7 segmentos, os pins podem estar na parte de cima da macro.

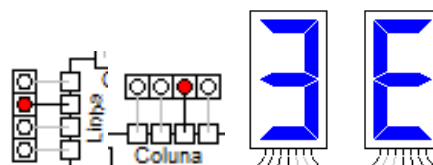
Teste manual: incluir no mínimo 4 cartas distintas decodificadas

Alínea B (1 valor)

Fazer um componente que permita ao utilizador seleccionar uma de 4 linhas, e uma de 4 colunas, e mostre ao utilizador a carta nessa posição. Em cada posição há uma carta distinta, a qual está num componente de memória de 16 posições. Carregar um dos mapas de memória com as 16 cartas baralhadas, fornecidos com este enunciado (ficheiros efolioA0.map a efolioA15.map).



Os dois bits menos significativos indicam o número da carta, e os dois bits seguintes o naipe da carta. Exemplo: na figura anterior a posição de memória 2 com os bits 1000, contém a carta de naipe 10b e número 00b, ou seja, 2C). As 16 posições de memória devem ser consideradas em matriz, com 2 bits de endereço menos significativos correspondendo a uma de 4 linhas, e 2 bits mais significativos correspondendo a uma de 4 colunas. Exemplo: a posição seleccionada na figura, a posição 9, corresponde a 1001b, tem a linha 01b seleccionada, e a coluna 10b seleccionada (2ª linha e 3ª coluna). A carta lá constante é a 1111b corresponde a 3E:



Assumir a ordem de escrita, ou seja, a primeira linha está em cima, e a primeira coluna à esquerda. Caso não exista uma linha ou uma coluna seleccionada, não deve ser mostrada nenhuma carta.

Circuito Combinatório:

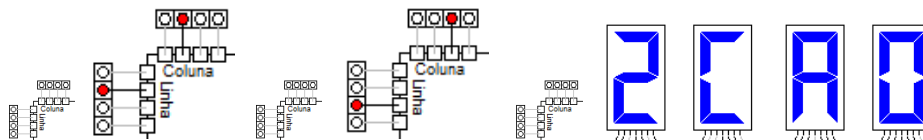
- Entradas:
 - Linha: 4 bits
 - Coluna: 4 bits
- Saídas:
 - Número: 7 bits
 - Naipe: 7 bits

Teste manual: incluir no mínimo 4 posições distintas, em termos de linhas e colunas, utilizando o ficheiro de memória efolioA0.map

Alínea C (1 valor)

Fazer um circuito sequencial em que escolha em sequência duas cartas, sem revelar a primeira carta até que a segunda seja escolhida. Após a escolha de ambas as cartas, estas devem ser mostradas em simultâneo, até que nova carta seja escolhida.

Exemplo:



No exemplo partimos de uma posição inválida, e colocamos a posição (2,2). De seguida colocamos uma posição inválida, após o qual colocamos a posição (3,3). Finalmente ao colocarmos uma posição inválida, vemos as cartas nas duas posições selecionadas, (2,2) correspondendo à posição de memória 0101b=5, e a posição (3,3) correspondendo à posição de memória 1010b=10. Nestas posições está o conteúdo de memória 1010b e 0100b, correspondendo às cartas 2C e A0.

Circuito Sequencial:

- Entradas:
 - Linha: 4 bits
 - Coluna: 4 bits
 - Reset: 1 bit - volta ao passo 1
- Saídas:
 - Número/Naípe carta A: 7+7 bits
 - Número/Naípe carta B: 7+7 bits

Sequencia:

1. Escolha inválida
2. Escolha válida - carta A é que está na posição escolhida
3. Escolha inválida
4. Escolha válida - carta B é a que está na posição escolhida
5. Escolha inválida - mostrar cartas A e B
6. Assim que exista uma escolha válida, ir para passo 2

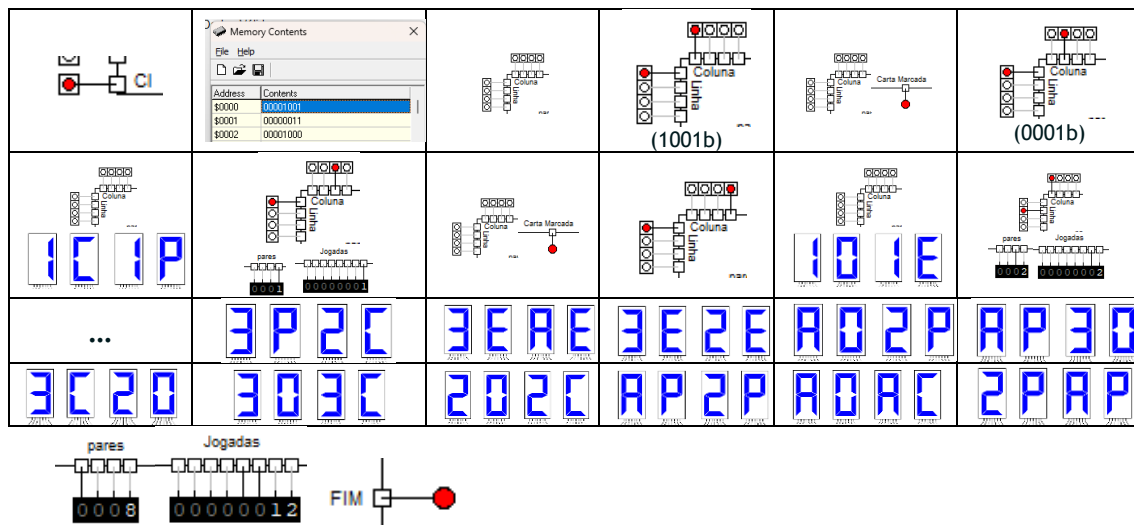
Teste manual: incluir no mínimo 4 seleções distintas, em termos de linhas e colunas, utilizando o ficheiro de memória efolioA0.map

Alínea D (1 valor)

Fazer um circuito sequencial, em continuidade do realizado na alínea C, sendo adicionadas as seguintes funcionalidades:

- Após selecionar a primeira carta, indicar que existe uma carta selecionada, até que seja selecionada a segunda carta
- Incrementar um contador cada vez que são selecionadas duas cartas, e mostrar o contador.
- Após mostrar as cartas, se tiverem naipe ou número igual, remover as cartas de modo a não mais mostrar as cartas. Exemplo: 1P e 2P mostradas, tem o naipe igual a P, pelo que são removidas.
- As posições com cartas removidas, se forem selecionadas, a seleção não é aceite, sendo necessário selecionar novamente.
- Contar o número de pares removidos, e mostrar esse contador. Assim que o contador chegue a 8, terminar o jogo e sinalizar o fim do jogo. Não há mais movimentos após o fim do jogo.
- Em qualquer altura, o *reset* repõe o sistema no passo 1. Para repetir o jogo, tem que se recarregar o conteúdo da memória para um dos baralhos de cartas (ficheiros efolioA0.map a efolioA15.map)

Exemplo:



Notas:

- Após carregar no play e reset, é que deve carregar a memória para assegurar o estado inicial
- Alerta-se que a ordem dos pins da coluna, é o inverso do normal, em que o menos significativo é à direita
- É necessário registo das posições das cartas removidas, de modo a não permitir novamente a sua seleção
- Os ficheiros *map* fornecidos, podem ser utilizados em memórias de 16 endereços, podendo ter mais de 4 bits de dados

Circuito Sequencial:

- Entradas:
 - Linha: 4 bits
 - Coluna: 4 bits
 - Reset: 1 bit - volta ao passo 1
- Saídas:
 - Número/Naipes carta A: 7+7 bits
 - Número/Naipes carta B: 7+7 bits
 - Fim: 1 bit
 - Carta marcada: 1 bit
 - Pares: 4 bits - pode ser visualizado com “Numeric Output”
 - Jogadas: 8 bits - pode ser visualizado com “Numeric Output”

Sequência: análise efetuada pelo estudante

Teste manual: incluir no mínimo 1 jogo completo, utilizando o ficheiro de memória efolioA0.map. Ao contrário do que ocorre no exemplo do enunciado, tem de indicar todas as seleções de linha/coluna realizadas durante o jogo.

NOTA: Crie de raiz todos os circuitos, usando os componentes digitais constantes da barra do Digital Works. Não utilize circuitos já feitos, exceto os realizados durante as atividades formativas, e não partilhados com terceiros.

BOM TRABALHO!

Regras a observar no desenho de circuitos:

Linhas conectoras sempre na horizontal e vertical e nunca na diagonal ou curvas.

Preferencialmente as entradas deverão aparecer no cimo ou lado esquerdo de cada componente, enquanto as saídas devem ser colocadas no fundo ou no lado direito dos componentes.

Regras para entrega do e-fólio A:

Forma de entrega:

Um ficheiro zipado com o nome correspondente ao número de aluno.

O ficheiro zipado deverá conter:

- Relatório em formato pdf de até 5 páginas A4, com todos os cálculos e todas as opções tomadas na construção do projecto, e
- Um ficheiro de Digital Works para as alíneas de a) a d), cujo nome seja o número de aluno mais a letra da alínea (ex: 999999c.dwm). As macros utilizadas são incluídas dentro do ficheiro .dwm que as utilizem.

Não são aceites entregas fora da plataforma Moodle.

Avaliação

Cotação: um valor por alínea

Critérios de Correção:

Funcionalidade: 50%

Modularidade da Solução (utilização de macros): 10%

Apresentação dos circuitos: 10%

Relatório (Legibilidade e Justificação dos Resultados e das Opções): 30%

Descontos:

Trabalhos entregues que não estejam em conformidade com as regras de entrega do e-fólio A: até 10%

Trabalhos entregues sem utilização do Digital Works: 50%

Deteção de fraude (total ou parcial): 100%

Trabalhos entregues após a data limite: 10% (o recurso de entrega estará aberto até ao início da manhã do dia seguinte, no entanto entregas após a data/hora estabelecida no PUC, têm desconto na nota).