

Relatório do projeto de Introdução à Arquitetura de Computadores

Turno: IAC4L10

Grupo: 72

Alunos:

- Rodrigo Bernardo, 78942

- Miguel Ribeiro, 79085

- Pedro Torres, 78742

Descrição do projeto

Este projeto consiste no desenvolvimento de um jogo de corridas de bicicleta, em que o objetivo é percorrer a maior distância possível sem chocar contra obstáculos. A bicicleta e o cenário envolvente, assim como os obstáculos, são escritos numa janela de texto e é aí que decorre o jogo. O movimento da bicicleta é desencadeado pressionando-se os botões de interrupção da placa P3 e o programa é desenvolvido tendo em vista a utilizar outros recursos disponibilizados pela placa, como o display de sete segmentos, o LCD e os LEDS.

Organização do programa

O código do programa está dividido em várias zonas:

1. Zona de registos, onde se refere qualquer utilização específica de um certo registo.
2. Zona de constantes (EQU), onde estão definidas todas as constantes.
3. Zona de palavras (WORD), onde estão definidas todas as palavras.
4. Zona de strings (STR), onde estão definidas todas cadeias de caracteres.
5. Zona de tabelas (TAB), onde estão definidas todas as tabelas.
6. Zona da tabela de interrupções.
7. Zona das interrupções, onde estão definidas todas as rotinas de interrupção.
8. Zona de rotinas, onde estão definidas todas as rotinas utilizadas pelo programa que não são de interrupção.
9. Programa principal, onde estão o código cujo seguimento se entende por ser o jogo propriamente dito.

Aspetos relevantes da implementação

Há vários pontos que são importantes de referir em relação à forma como o nosso programa foi implementado:

1. Escrita da pista, da bicicleta e dos obstáculos e dos objetos em geral

Todos os objetos são escritos com base em rotinas de escrita de cadeias de caracteres. Isto torna a realização do código das rotinas muito simples. No entanto, deve ser apontado o seguinte: na zona STR pode-se ver a definição de strings como “IIIIIIIIIIIIIIIIIIII”, que corresponde a uma parte de uma parede da pista. Esta forma permite escrever a pista de uma forma muito simples

recorrendo apenas a uma rotina que escreva numa coluna da janela; esta implementação permite também a escrita fácil de paredes com vários caracteres distintos.

2. Rotina TURN_BIKE (do movimento da bicicleta)

Por vezes, no simulador (e apenas no simulados) são geradas “múltiplas” bicicletas quando se carrega muito rapidamente nos botões de interrupção que permitem o movimento. Isto deve-se ao seguinte: atendendo à estrutura da rotina, se se carregar num desses botões enquanto a bicicleta está a ser escrita na janela, o valor colocado na posição BIKE_POS não será o da bicicleta escrita. Como o valor da posição BIKE_POS é utilizado pela rotina que apaga a bicicleta, esta estará a apagar num sítio errado, gerando-se múltiplas bicicletas aquando de novo movimento. Esta situação não ocorre (ou é altamente improvável que ocorra) na placa devido à velocidade com que processadas as instruções.

Este problema é facilmente resolvido inibindo-se as interrupções dentro da rotina (DSI/ENI), mas decidimos não o fazer após conversação com um dos professores.

3. Rotina LEVEL_UP (da subida de nível)

É de referir que entramos no nível 1 sempre que a rotina é chamada, mas, mesmo se for suposto estar num nível superior, ou seja, mesmo se não quisermos baixar o nível, a velocidade a que ocorrem as instruções não permite notar qualquer alteração (quer na placa, quer no simulador).

4. Rotina RESET_BOARD (para apagar a janela)

Esta rotina apaga toda a janela de texto. Na placa, como o processamento das instruções é muito rápido, não tem qualquer problema. No entanto, se o jogo tivesse sido desenvolvido com vista a ser jogado no simulador, a abordagem deveria ser diferente, pois esta rotina demora muito tempo a processar completamente no simulador, tornando a experiência de jogo desagradável.

5. Interrupções TURN_LEFT e TURN_RIGHT (do movimento da bicicleta)

A forma como estas interrupções estão estruturadas permite mover a bicicleta uma posição por cada vez que se pressiona os botões de interrupção IO e IB. Se tivéssemos desenvolvido as interrupções para que estas ativassem um indicador de movimento, mesmo que pressionássemos mais que uma vez os botões, a bicicleta só se moveria uma vez. No entanto, como a placa processa as instruções muito rapidamente, isto não se notaria.

Conclusão

Para concluir, deve-se referir que o desenvolvimento do jogo está finalizado, como planeado, sem nada a apontar ao seu funcionamento (pelo menos até agora). O programa inclui também as versões avançadas turbo e pausa. No entanto, a nossa versão pausa difere da do enunciado, no sentido de que escrevemos a mensagem de pausa no ecrã, em vez de no LCD. Desta forma, é mais perceptível que o jogo se encontra pausado e é possível ler as distâncias no LCD. De

qualquer forma, no ficheiro .zip fornecido está incluído um PDF com uma versão da rotina de pausa em que a mensagem de pausa é escrita no LCD em vez de na janela.

Por último, referimos que, se pudéssemos voltar atrás, teríamos desenvolvido o programa de forma a não especificar o registo R7 como guardador da posição atualizada da bicicleta, mas sim a determinar uma posição de memória para esse efeito, pois seria mais lógico (embora o programa não tenha nada a perder com a forma como foi implementado).