**Introdução à Arquitetura De Computadores**

**Relatório do Projeto**

**Introdução**

No seguimento da disciplina foi nos proposta a realização de um projeto que consiste no jogo “TRON”, em linguagem Assembly. Para a realização do projeto foram criadas rotinas que utilizam o display de 7 LEDS, LCD, display de 16 LEDs, sendo todos os inputs de dados e as passagens de tempo são processados por rotinas de interrupção.

**Estrutura do programa**

O programa pode ser dividido em três partes, nomeadamente, o início do programa, o jogo em si e o fim do jogo

**1ª Parte - Início**

Essencialmente destinada às diversas inicializações, contém um ciclo que é executado até o utilizador premir a interrupção 1. É também nesta fase que o utilizador pode ativar o modo *Hardcore*,

As rotinas mais importantes desta parte do programa são:

**Inicializar** – rotina que inicializa a pilha e o cursor, limpa o LCD, e ativa a interrupção I1.

**Texto\_inicial** – rotina que escreve o texto inicial “Bem-vindo ao TRON. Pressione I1 para comecar.”

**2ª Parte – Jogo**

Parte principal do jogo, na qual se desenha o campo de jogo, as partículas, do seu movimento e são detectadas as colisões.

Rotinas mais importantes:

**Jogo** – rotina que desenha o campo de jogo, as partículas na sua posição inicial, reinicia as direções das partículas e inicializa o temporizador.

**Partículas** – verifica as direcções (compara as próxima direção com a direção atual, se esta for maior, incrementa a direcção actual, se for menor decrementa a direção atual, e se for igual mantém a direcção, isto de modo a precaver o caso de o utilizador premir várias vezes a mesma interrupção. Assim a representação das direções adotada por nós foi 1,2,3 e 4, correspondendo estas a cima, direita, baixo e esquerda, respetivamente) para as quais as partículas se vão mover, verifica se há colisões e só após estas verificações move de facto as partículas, não apagando o seu rasto.

**IngameLoop** – loop que controla todo o funcionamento do jogo. É a partir deste loop que se chamam as todas as rotinas que operam durante o decorrer do jogo, e é também a partir desta rotina que partem os comandos para escrita na placa.

**3ª Parte – Fim do Jogo**

Após o fim do jogo, a pontuação e o tempo máximo são actualizados, e surge o texto para reiniciar o jogo (premindo I1).

Rotinas mais importantes:

**ReiniciaJogo** – esta rotina escreve o texto final, reinicia os contadores do temporizador, reinicia o tempo do display de 7 segmentos repõe as posições iniciais das partículas, o valor da velocidade e as direcções. Depois reinicia as interrupções e espera que o utilizador pressione I1 de novo para recomeçar o jogo.

**Funções adicionais**

**Pausa** – conseguida através de um loop, na rotina **VerificaPausa.** A pausa encontra-se associada ao interruptor 7. Nesta rotina as direcções são salvaguardadas, de modo a que se forem premidas as interrupções as direcções voltam ao seu valor original.

**Modo HARDCORE** – neste modo não existem níveis, logo o jogo é sempre jogado à velocidade máxima (nível 5) e não é possível pausar o jogo.

**Manual do Jogo – TRON**

**Objetivo do jogo**

O jogo TRON tem como objectivo bloquear o avanço do adversário, ao mesmo tempo que se procura avançar sem ser bloqueado.

**Controlos**

**Movimento**

**Jogador 1**

I0 – move partícula para a esquerda

IB – move partícula para a direita

**Jogador 2**

I7 – move partícula para a esquerda

I9 – move partícula para a direita

**Outros**

I1 – Começa/recomeça o jogo

Interruptor 0 – Modo HARDCORE

Interruptor 7 – Interrompe o jogo

**Conclusão**

Neste projeto conseguimos atingir os objectivos, tendo conseguido desenvolver e implementar todas as principais funções do programa.

Adicionalmente também foram implementadas funcionalidades extra, nomeadamente a possibilidade de parar/recomeçar o jogo, assim como a existência de um modo HARDCORE, descrito previamente.

Um dos aspetos do programa a melhorar seria a nível da reserva de espaços na matriz. A nossa matriz estaria mais eficiente se tivéssemos implementado a opção adicional de alterar o tamanho da janela de jogo. Poderíamos também ter melhorado a organização das rotinas no projeto bem como a salvaguarda de registos. No entanto isso não interfere diretamente com a execução do programa.

**Grupo 30:**

Inês Oliveira, nº 78991 Pedro Cerejo, nº 81338 João Oliveira, nº 81670