

Introdução à Arquitetura de Computadores
Jogo” TRON”

Filipe Azevedo, nº82468

Martim Silva, nº82517

Ana Nogueira, nº82433

LEIC

5 de Dezembro de 2014

Prof. Nuno Horta



Objectivos

O projecto da cadeira de Introdução à Arquitectura de Computadores tem como objectivo a consolidação de conhecimentos sobre programação em Assembly do processador P3, pretendendo-se implementar o jogo “Tron”, fazendo recurso ao simulador utilizado durante as aulas laboratoriais semanais. O jogo consiste na movimentação de duas partículas numa área de jogo de 20 linhas por 48 colunas, delimitada horizontal e verticalmente, sem que colidam uma com a outra ou com os rastros já definidos pelas mesmas. Para além do funcionamento básico do jogo, são ainda implementadas outras funcionalidades na janela da placa do simulador: o LCD apresenta o número de vitórias de cada jogador e o tempo máximo de jogo; o conjunto de 16 LEDS acende em grupos de quatro, indicando quais os níveis já completados; os 4 displays de 7 segmentos fazem a contagem do tempo de jogo; por último, os interruptores exteriores I0 e IB servem para movimentar a partícula do primeiro jogador e os interruptores exteriores I7 e I9 servem para movimentar a partícula do segundo jogador.

Ciclo de Execução do Programa

Após a iniciação do programa a partir do endereço de memória 0000h, é efectuado um salto (JMP) para a rotina Inicio3, que corresponde ao local onde todas as rotinas serão chamadas. Em ‘Inicio3’ é efectuada a iniciação do *stack pointer* com o valor FDFFh, é feita a activação da máscara de interrupções, inicializado o temporizador e inicializados os displays de sete segmentos com o valor zero. Depois chama-se a rotina ‘Inicio1’ que escreve a mensagem de texto ‘Bem vindo ao Tron / Prima o Interruptor I1 para começar’ no ecrã de jogo. Chama-se de seguida a rotina ‘ComecaLCD’ que efectua a escrita das strings iniciais no LCD, que é chamada apenas no início do jogo. O ecrã vai conter esta mensagem até se premir o interruptor I1 que aciona a rotina de interrupção ‘ecra’, que põe a flag ‘LoopJogo’ a 1, iniciando o jogo. A partir do momento que é premido o interruptor I1, são chamadas três rotinas, ‘Apaga’, ‘ApagaTab’,

'Moldura' em que todas juntas resultam em que seja criada o espaço de jogo com o tabuleiro, limpando através da 'ApagaTab' a memória do espaço de jogo e através da 'Apaga' limpa-se os caracteres do jogo anterior que se encontravam no espaço de jogo. De seguida, chamam-se estas rotinas: MemContSup, MemContInf, MemContDir MemContEsq, que escrevem numa tabela em memória as posições na janela. Para se seleccionar a direção de cada um das partículas, decidiu-se escolher um número para cada uma das quatro direções: 1 correspondente à direção 'cima', 2 correspondente à direção 'direita', 3 correspondente à direção 'baixo' e 4 correspondente à direção 'esquerda'. Escolheu-se o R5 para guardar a direção da partícula do primeiro jogador ('X') e R6 para guardar a direção da partícula do segundo jogador ('#'). Como o primeiro jogador começa o jogo na direção 'baixo', inicialmente, o Xinicial é igual a 3 e ao carregar no interruptor I0, decrementa-se o R5, sendo a nova direção de 'X', a direção 'direita'. Ao carregar no interruptor IB, incrementa-se o R5 e, no caso de o 'X' estar a movimentar-se para a direita (2), o 'X' passa a movimentar-se para 'baixo'. No caso da partícula de jogador 2 ('#'), o mecanismo é o mesmo, sendo utilizado o R6 para guardar a direção e os interruptores I7 e I9 para decrementar e incrementar, respetivamente. De seguida, coloca-se os displays e os LEDS a zero. A seguir coloca-se no R1 a posição inicial em que se vai escrever o 'X' e no R2 a posição inicial em que se vai escrever o '#'. No R4 vamos guardar a quantidade de décimas de segundo que passam enquanto se está a jogar, de forma a aumentarmos de nível e no R7 guarda-se as décimas de segundo que passam durante o jogo, de forma a incrementar os displays de 7 segmentos. O 'Ciclo7' vai ser o ciclo principal do jogo para onde salta o código sempre que há a escrita de um carater. No início do ciclo faz-se a chamada para a rotina de pausa que verifica se o botão de pressão 7 está ativo. De seguida inicializa-se o R3, registo que vai servir para fazer a contagem das décimas de segundo que irá servir para fazer andar as partículas de acordo com o nível de jogo. Seguidamente compara-se o R4 com o tempo de jogo em que é necessário passar para o próximo nível. Quando sabe em que nível vai, faz-se um JMP para a etiqueta corresponde ao nível em que se chama a rotina escreveLed correspondente, caso

seja necessário, o 'Delay' (rotina que espera durante 1 décima de segundo), incrementa-se o R7, Incrementa-se o R4 e chama-se a rotina 'Displays', de forma a atualizar os displays de sete segmentos. Depois de se esperar o tempo necessário, faz-se um JMP para a rotina 'escreve' em que se faz um CALL das rotinas ContornoFimS, ContornoFimI, ContornoFimD, ContornoFimE que verificam se não houve colisão das partículas com os contornos do espaço de jogo e CALL da rotina PercorreTab que verifica se não houve colisão com outras partículas já impressas. Seguidamente coloca-se o R1 e R2 (posição em que se vai escrever as partículas na janela de texto) no porto de controlo e fazemos CALL às rotinas que escrevem as partículas na janela de texto. A seguir existe um CALL das rotinas AndaX e AndaC, que verificam qual a próxima direção da partícula e, por fim, faz-se um JMP para o Ciclo7 (ciclo principal de jogo).

Conclusão

A realização do projeto correu como planeado, tendo sido cumpridos todos os objectivos delineados.

Consideramos que este projeto foi extremamente importante na consolidação do conhecimento adquirido nas aulas teóricas e no desenvolvimento do raciocínio lógico inerente ao desenvolvimento de código Assembly, foi também um desafio bastante interessante em que pôs o nosso pensamento abstracto em acção e consolidou as nossas competências de trabalhar em grupo devido à grande organização do trabalho tão complexo como este.

De modo a melhorar a experiência de jogo foi implementada uma função adicional de pausa, que é activada quando se pressiona o botão de pressão 7.