

INTRODUÇÃO À ARQUITETURA DE COMPUTADORES

Relatório do projeto
“Pássaro Bamboleante”

GRUPO 50

83418- AFONSO FEIJÃO
83431 ANDREIA LOPES
83550 PEDRO SANTOS

Introdução

Na disciplina de Introdução à Arquitetura de Computadores (IAC), foi-nos dado como projeto a realização do jogo “Pássaro Bamboleante”. Este será programado e codificado em linguagem Assembly do processador P3.

Resumidamente, o jogo consiste em, num cenário fechado, ultrapassar o maior número de obstáculos (portas), havendo três níveis de velocidade integrados que podem ser controlados pelo utilizador.

O jogo está dividido em 3 partes essenciais:

1. Inicialização
2. Jogo
3. Fim de Jogo

1.Inicialização:

- Mensagem inicial. O jogo é iniciado quando o botão I1 for premido.

2.Jogo:

- Movimentação dos tubos da direita para a esquerda (a velocidade pode ser controlada pelo uso das interrupções I1 e I2. A interrupção I1 diminui a velocidade de jogo e a I2 aumenta);
- Implementação de um ciclo aleatório, para gerar uma posição da porta aleatória;
- Implementação do LCD, display de 7 segmentos, para transmitir ao jogador o número de obstáculos ultrapassados;
- Implementação de LEDS par indicar o nível de jogo (4 níveis);
- Implementação do display LCD, que mostra a distância percorrida.

3.Fim de Jogo:

- O jogo termina quando o pássaro embater num obstáculo ou no limite inferior;
- Implementação de uma mensagem de fim de jogo.

Rotinas

Foram implementados 4 **níveis** de velocidade: **NÍVEL 1**- 4 LEDS ; **NÍVEL 2**- 8 LEDS ; **NÍVEL 3**- 12 LEDS ; **NÍVEL 4**- 16 LEDS. Os leds são atualizados quando é detetada uma interrupção ativa (I1 ou I2);

A **gravidade** foi implementada com recurso a um contador, cujo valor varia ao longo do tempo. O valor do contador é decrementado progressivamente enquanto a interrupção I0 (saltar) não é premida, criando um movimento acelerado;

Para a movimentação dos tubos foi usada uma tabela com 13 posições. Nos dois primeiros bits foi guardada a linha da primeira posição livre da porta (valor gerado pela função aleatória) e nos dois últimos bits foi guardada a posição atual do tubo (coluna do tubo).

Existem quatro rotinas principais para a movimentação dos tubos:

- AvancaTubos (Rotina principal)**: percorre as 13 posições da tabela, reinicia o valor se um tubo chegar ao início, atualiza as posições da tabela e chama as funções que escrevem e limpam os tubos;

- CriaTubos**: inicializa as posições da tabela com o valor inicial (posição porta, posição tubo inicial);

- Limpatubo**: limpa o tubo;

- AvancaTubo**: atualiza a posição do cursor e escreve o tubo.

A **rotina aleatória** gera a linha da primeira posição livre da porta. É usada uma divisão por doze, porque foi implementada uma “margem de segurança” de 3 posições (superior e inferior) onde não podiam aparecer tubos.

As **colisões** foram implementadas verificando cada posição da tabela. Se o tubo se encontrar na coluna do pássaro, são verificadas as linhas. Se o pássaro NÃO se encontrar nas linhas da porta então significa que colidiu com a coluna.

As rotinas do **display de 7 segmentos** e as rotinas do **LCD** são chamadas de outras rotinas e não do programa principal poupando, assim, processamento já que só são invocadas quando é necessário atualizar os seus valores.

Estruturas de dados

Foi utilizada uma tabela de 13 posições para guardar os valores necessários para a movimentação dos tubos. Todas as outras **variáveis** são essencialmente flags ou ponteiros para escrita na janela de texto.

Conclusão

Ao longo da realização deste trabalho fomos adquirindo conhecimentos acerca da linguagem Assembly do P3. Ao início deparamo-nos com algumas dificuldades, principalmente ao nível da sintaxe do P3, devido à característica de baixo nível desta linguagem.

O nível de exigência para a realização do projeto deveria, na nossa opinião, ter sido um pouco mais equilibrado relativamente às aulas teóricas.

O projeto foi de um modo geral aliciante, tendo contribuído para o desenvolvimento de métodos e técnicas de programação.