

ARQUITETURA DE COMPUTADORES

UALG – ISE – LESTI

Vitrine Natalícia

56278 – Jose Victor Santos – a56278@ualg.pt
71433 – Gonçalo Marques – a71433@ualg.pt
79299 – Diogo Rodrigues – a79299@ualg.pt

1. Manual de utilizador

Este manual descreve as etapas para configurar e utilizar o projeto final da cadeira de Arquitetura de Computadores, que tem como objetivo aplicar os fundamentos da linguagem de programação de baixo nível “Assembly”, com ênfase na interação entre a CPU (PEPE-16) e os periféricos, simulados digitalmente através do simulador SIMAC (v2.3.0). O projeto, intitulado "Vitrine Natalícia", proporciona uma oportunidade prática para consolidar os conceitos aprendidos ao longo da disciplina.

2. Introdução

O projeto simula uma vitrine natalícia interativa, que combina elementos visuais e sonoros, controlados pelo utilizador através de um teclado. O cenário busca integrar hardware e software para criar uma experiência festiva.

3. Passos para configuração e execução da vitrina natalícia

3.1. Configurações realizadas no circuito digital

- O circuito digital utilizado foi fornecido pelo docente (projeto-vitrine-natalicia.cir), cujos componentes já haviam sido previamente conectados e parcialmente configurados no simulador SIMAC (v2.3.0), nomeadamente:
 - Processador PEPE-16
 - Decodificador
 - Memória RAM
 - MediaCenter
 - Periféricos de entrada/saída (PIN, POUT-1 e POUT-2)
 - *Display* de sete segmentos
 - Teclado
 - Operador lógico AND

- Para realização deste trabalho, então, carregamos o circuito no SIMAC, selecionando a opção "Load Design" e escolhendo o ficheiro "Cena-1\projeto-vitrine-natalicia.cir".
- Em seguida, carregamos o ficheiro contendo o código Assembly a ser executado pelo processador PEPE-16, fazendo duplo clique sobre este componente na *circuit área*, e selecionando o ficheiro "Cena-1\vitrine_natalicia.asm" no campo *program file*.
- No passo seguinte, configuramos o componente MediaCenter, onde fizemos as seguintes alterações:
 - *Image and test pixel patterns:*
 - *Pixel screens:* optamos por desenhar cada objeto da cena em uma camada diferente, para facilitar a manipulação dos objetos individualmente, facilitando, por exemplo a ação de mostrar ou ocultar um objeto através de comandos. Portanto, definimos um total de 8 *pixel screens* (0-7).
 - Ficheiros de imagem: carregamos um total de 4 imagens a serem utilizadas como diferentes imagens de fundo da cena.
 - *Sounds vídeos and gif:*
 - Ficheiros de áudio: carregamos um total de 8 ficheiros de áudio utilizados como som de fundo ou efeito sonoro associado a animações. Para os ficheiros mais curtos, até 3s, selecionamos a opção "*as clip*", para que fique carregado em memória e seja executado com menor latência, conforme disposto na documentação do MediaCenter.

3.2. Execução do projeto no SIMAC

- Para execução do projeto Vitrine Natalícia, comece por descarregar e descompactar a pasta do trabalho AC_PF_LESTI_56278-71433-79299 para o seu computador.
- Inicie o simulador no sistema operativo em uso, seguindo o método padrão de inicialização, nomeadamente duplo clique no ficheiro executável correspondente à resolução mais adequada para o seu caso.
- Carregue o circuito "projeto-vitrine-natalicia.cir" através do método descrito na secção 3.1 (as configurações mencionadas anteriormente já estarão todas feitas).
- Em seguida, carregue na aba "*Simulation*" e então no botão "*Start*", na barra de ferramentas de simulação.
- Certifique-se de abrir os componentes PEPE-16, MediaCenter e Teclado, fazendo duplo clique sobre os mesmos, e de carregar no botão "Start" do PEPE-16 para iniciar a simulação da vitrine natalícia
- O controlo da cena deve ser feito através do teclado, cujas funcionalidades serão descritas adiante.

3.3. Controlo da cena com o teclado

Para controlo da cena da vitrine natalícia, o nosso código Assembly foi escrito de modo a garantir o uso completo do teclado (0H – FH, 16 teclas), conforme as funcionalidades abaixo especificadas:

Tecla	Funcionalidade
0	Liga/desliga o objeto <i>giftbox</i> e reproduz efeito sonoro associado
1	Liga/desliga o objeto <i>pai_natal</i> e reproduz efeito sonoro associado.
2	Liga/desliga o objeto <i>arvore</i> e reproduz efeito sonoro associado.
3	Liga/desliga o objeto <i>merry_xmas</i> e reproduz efeito sonoro associado.
4	Para a reprodução de todos os sons em execução (stop).
5	Reproduz música de fundo (ficheiro de áudio 0).
6	Reproduz música de fundo (ficheiro de áudio 1).
7	Reproduz música de fundo (ficheiro de áudio 2).
8	Mostra a imagem de fundo (ficheiro de imagem 0).
9	Mostra a imagem de fundo (ficheiro de imagem 1).
A	Mostra a imagem de fundo (ficheiro de imagem 2).
B	Mostra a imagem de fundo (ficheiro de imagem 3).
C	Liga as duas animações simples (neve e luzes da árvore a piscar individualmente com uso de temporizador não bloqueante), reproduzindo uma música de fundo associada, específica da animação simples. Note-se que se esta animação estiver ativada e o utilizador carregar na tecla 2, para esconder a árvore, as luzes de natal param de piscar e são ocultadas também, não afetando a animação da neve, que continua a ser reproduzida até que se carregue na tecla D.
D	Desliga as animações simples ativadas pela tecla C (se desejar parar o som associado à animação, deve usar a tecla 4).
E	Liga a animação complexa que move o letreiro <i>merry xmas</i> horizontalmente (ao mesmo tempo em que desativa as animações simples para melhor fluidez).
F	Desliga a animação complexa retornando o letreiro para a sua posição inicial.

4. Comentários

4.1. Realizações

O projeto atendeu a todos os objetivos das versões V0, V1 e V2, descritos no enunciado, que incluem:

- Integração de 6 elementos decorativos, desenhados em *pixel art* por nós:
 - Caixa de presente;
 - Pai natal;
 - Árvore de natal;
 - Letreiro *Merry xmas*;
 - Conjunto de 2 objetos diferentes para simular a neve a cair;
 - Conjunto de 2 objetos diferentes para simular luzes a trocar de cor.
- Utilização de imagens de fundo apropriadas:
 - Foram disponibilizadas 4 imagens de fundo.
- Implementação de animações simples e complexas com uso de temporizador e rotinas cooperativas/não bloqueantes:
 - Animação em que dois objetos (em duas *pixel screens* diferentes) alternam a sua exibição, de forma temporizada, para dar o efeito de neve a cair;
 - Animação em que dois objetos (em duas *pixel screens* diferentes) alternam a sua exibição, de forma temporizada, para dar o efeito de luzes a piscar e trocar de cor;
 - Animação complexa do letreiro *merry xmas*, que se move horizontalmente no ecrã, de forma temporizada e não bloqueante;
 - Possibilidade de exibir ou ocultar os elementos decorativos através do teclado.
- Efeitos sonoros:
 - Possibilidade de escolher 4 sons de fundo diferentes;
 - Efeitos sonoros associados a cada elemento decorativo (são reproduzidos ao exibir o objeto – teclas 0, 1, 2 e 3);
 - Som de fundo exclusivo reproduzido apenas quando é iniciada a animação da neve e das luzes de natal.
- Teclado:
 - Teclado completamente funcional, com as funcionalidades já descritas na subsecção 3.3.

Todos os elementos especificados no manual de utilizador funcionam conforme esperado, evidenciando uma integração bem-sucedida entre o código Assembly e os periféricos simulados.

No código Assembly, utilizamos rotinas cooperativas genéricas, sempre que possível, para uma melhor eficiência, modularização e reutilização de código.

A rotina para detecção das teclas do teclado faz o varrimento em ciclo das quatro linhas do teclado e detecta a tecla premida, obtendo o seu valor hexadecimal (0 a FH), em vez dos valores separados da linha e coluna (que havia sido a nossa primeira abordagem para implementação do teclado 4x4).

Para uma melhor organização e documentação temos o código todo comentado e separado, visualmente, por separadores horizontais. Além disso, usamos o GitHub como ferramenta de trabalho colaborativo para potencializar o trabalho em equipa.

4.2. Limitações

Apesar do sucesso alcançado, algumas funcionalidades que poderiam elevar o nível técnico e a interatividade do projeto não foram implementadas devido a restrições de tempo ou limitação do simulador/componente. Entre elas:

- a) **Maior controlo da execução dos sons de fundo:** inicialmente pretendia-se que a funcionalidade de reproduzir um som de fundo (teclas 5, 6, e 7), primeiro parasse um som de fundo que já estava a ser reproduzido anteriormente, para então iniciar a reprodução do novo som de fundo, correspondente à tecla carregada. No entanto, os comandos do MediaCenter para parar a reprodução de um som específico ou de todos sons (66H e 68H, respetivamente), mostraram-se problemáticos, pelo que se optou por não parar a reprodução de um som atual automaticamente, cabendo ao utilizador fazê-lo com a tecla 4 (ou então terá os sons a serem reproduzidos simultaneamente).
- b) **Fluidez das ações das teclas quando usadas como interruptor:** notamos que ao utilizar as teclas do teclado como um interruptor para ligar/desligar alguma animação ou objeto consoante o seu estado atual, este tem tendência a piscar e o controlo acaba por não ser muito preciso, como acontece com as teclas 0, 1, 2 e 3. Enfrentamos este mesmo problema com as animações simples, quando tínhamos teclas distintas para controlar a animação da neve e das luzes (a tecla C ligava/desligava a neve e a D as luzes), por isso optamos por usar uma tecla para ativar a animação de ambas e outra para desliga-las, tendo assim o problema ficado resolvido para as animações.
- c) **Execução baseada em processos cooperativos e mecanismo de interrupções:** uma abordagem que permitiria uma gestão mais eficiente das animações simultâneas (especialmente animações complexas).
- d) **Uso de *display* para indicação da cena atual e da tecla pressionada:** um display informativo que melhoraria a usabilidade e tornaria o projeto mais intuitivo.
- e) **Múltiplas cenas e animação em dois eixos:** vitrine com três cenas festivas diferentes, pelo menos 10 objetos e animação complexa com movimento em dois eixos.

4.3. Sugestões de Melhorias

4.3.1. Implementação de Interrupções e Processos

- A introdução de interrupções permitiria melhorar o desempenho do projeto, assegurando uma execução mais eficiente e responsiva, especialmente durante animações complexas
- A criação de processos paralelos simplificaria a gestão das diferentes animações e elementos sonoros, tornando o sistema mais modular e escalável.

4.3.2. Implementação de mais opções de cenas, objetos e animações

- Desenvolver três cenários distintos, cada um com uma combinação única de animações, sons e elementos visuais. Isso enriqueceria a experiência oferecendo maior diversidade e personalização.
- Expandir o número de elementos decorativos disponíveis para incluir uma variedade maior de temas natalícios, como luzes pisca-pisca, renas animadas ou presentes, permitindo composições mais complexas e criativas.
- Criar animações mais sofisticadas, onde os objetos decorativos podem se mover simultaneamente tanto na horizontal como na vertical. Isso traria maior dinamismo ao projeto, simulando efeitos como queda de neve em diferentes direções ou movimentos realistas de enfeites

4.3.3. Displays para Apresentação da Cena e Tecla Pressionada

- Adicionar um display no simulador para mostrar informações em tempo real, como a cena festiva atualmente exibida e a tecla pressionada. Isso tornaria a interação mais intuitiva e o sistema mais informativo.

4.3.4. Investigar possíveis melhorias dos itens a e b da subsecção 4.2

- Dedicar mais tempo a resolver as limitações relativas ao controlo dos sons de fundo e das ações das teclas 0, 1, 2 e 3.