UNIVERSIDADE DO VALE DO ITAJAÍ

ESCOLA DO MAR, CiÊNCIA E TECNOLOGIA

curso de Engenharia de Computação

miCROCONTROLADORES

Implementação do jogo asteiróide em um microcrontrolador Pic18F4520

por

Diogo Agenor Marchi

George de Borba Nardes

Itajaí (SC), Junho de 2020

Universidade do Vale do Itajaí

ESCOLA DO MAR, CIÊNCIA E TECNOLOGIA

Curso de Engenharia de Computação

miCROCONTROLADORES

Implementação do jogo asteiróide em um microcrontrolador Pic18F4520

por

Diogo Agenor Marchi

George de Borba Nardes

Relatório apresentado como requisito parcial da disciplina Microcontroladores do Curso de Engenharia de Computação para análise e aprovação.

Professor Responsável: Paulo Roberto Valim

(obs.: modelo baseado no usado na disciplina de Projeto de Sistemas Embarcados)

Itajaí (SC), Junho de 2020

Sumário

[1 INTRODUÇÃO 4](#_Toc41893273)

[1.1 Descrição do projeto 4](#_Toc41893274)

[1.2 Objetivo geral 4](#_Toc41893275)

[1.3 Delimitação do escopo 4](#_Toc41893276)

[1.4 Trabalhos semelhantes 6](#_Toc41893277)

[1.5 Análises de requisitos 6](#_Toc41893278)

[1.5.1 Requisitos funcionais 6](#_Toc41893279)

[1.5.2 Requisitos não funcionais 7](#_Toc41893280)

[1.5.3 Regras de negócio 7](#_Toc41893281)

**2 REFERÊNCIAS......................................................................................................................8**

1. INTRODUÇÃO
   1. Descrição do projeto

O projeto escolhido envolve o desenvolvimento do jogo de asteroide para um microcontrolador PIC. O objetivo do jogo é fazer com que a nave desvie dos asteroides que estiverem vindo em sua direção. Caso ocorra uma colisão com asteroide o jogo é finalizado. O controle da nave é feito por um GAMEPAD e o jogo é exibido em um display LCD.

Como o jogo será executado em um microcontrolador, algumas limitações deverão ser consideradas, como a falta de um SO para gerenciamento de entradas e saídas. Além disso, existe a necessidade de que seja considerado o funcionamento do jogo como um sistema executando threads em paralelo. Será feito um chaveamento entre funções para que o usuário tenha a impressão de que todas as funções do jogo estão sendo executadas paralelamente.

* 1. Objetivo geral

Este projeto de conclusão da matéria microcontroladores, tem como objetivo aplicar alguns conhecimentos adquiridos durante o semestre. O tema escolhido pelos alunos foi motivado por englobar alguns dos principais assuntos tratados, como integração do PIC18F4520 com o display HD44780, módulo de tratamento de interrupções externas e interrupções internas do PIC.

* 1. Delimitação do escopo

Será usado um display de 16 x 4(LM041L) como saída, e como entrada um keypad alterado para as necessidades do jogo. O keypad terá setas no eixo vertical e horizontal, uma tecla “enter” e uma tecla “escape”. O microcontrolador que será usado é o PIC18f4520. Todos os componentes podem ser encontrados no ambiente de simulação Proteus.

O diagrama de estado da Figura 1 apresenta todos os estados atingíveis durante a execução do jogo. Fica implícito que cada estado pode ser interrompido ao microcontrolador ser desenergizado.

A Figura 2 apresenta do diagrama de blocos do projeto. O bloco GAMEPAD é uma matriz de botões adaptada para as necessidades do jogo. O bloco DISPLAY LM041L será o display LCD no qual o jogo será exibido. Ambos os blocos terão as ações controladas pelo bloco Módulo Timer, que é interno ao PIC.

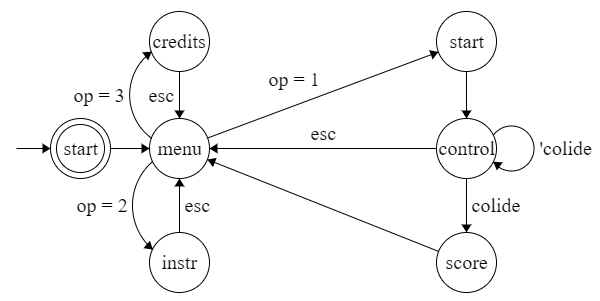


Figura 1- Diagrama de estados do jogo

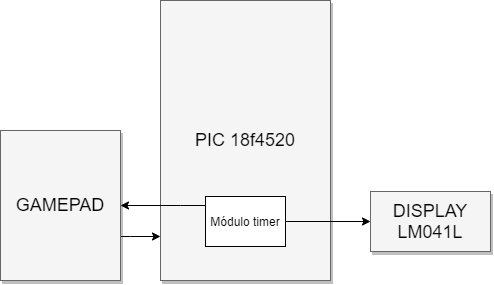


Figura 2- Diagrama de Bloco do Projeto

A saída do GAMEPAD é conectada as portas de entrada e saída do PIC. O estado das teclas serão verificados de tempo em tempo. O tempo será definido de forma que não afete o desempenho da exibição do jogo no display e não adicione delay no reconhecimento das ações do player. Será definido também um tempo de atualização do display LCD. O objetivo é fazer com que o usuário tenha a impressão de que as ações estão ocorrendo simultaneamente.

* 1. Trabalhos semelhantes

Foram feitas algumas pesquisas sobre jogo de Asteroide com o display LCD, e pelo que foi achado, existem jogos apenas com o microprocessador Arduino UNO ou MEGA, então, o asteroide com o PIC18f4520 seria o diferencial, e também, este projeto movimentará a nave nos dois eixos, eixo x e eixo y, em um display LCD 16x4.

Tabela 1– Comparativo da solução proposta com soluções existentes

|  |  |  |  |
| --- | --- | --- | --- |
| **Nome** | **Microcontrolador** | **Display** | **movimento** |
| [SanticN4N](https://www.instructables.com/member/SanticN4N/) | Arduino Uno | LCD16X2 | Sensor movimento |
| Este projeto | PIC18F4520 | LCD16X4 | Controle com botão |

* 1. Análises de requisitos
     1. Requisitos funcionais

Nesta seção, será apresentando os requisitos funcionais que definem as principais funções a serem realizadas pelo sistema. Cada requisito funcional é conferido por meio da aplicação de um plano de verificação elaborado.

* RF01: O sistema deve permitir que o usuário navegue pelo menu principal;
* RF02: O sistema deve permitir que o usuário selecione uma opção do menu principal;
* RF03: O sistema deve permitir que o usuário retorne ao menu principal a qualquer momento;
* RF04: O sistema deve armazenar as pontuações;
* RF05: O sistema deve incrementar a pontuação durante o jogo.
* RF06: O sistema deve permitir que o usuário movimente a nave no espaço.
* RF07: O sistema deve identificar uma colisão da nave com um asteroide.
  + 1. Requisitos não funcionais

Nesta seção será apresentado os requisitos não-funcionais que definem as principais restrições de implementação e características do sistema a ser desenvolvido. Estas características serão conferidas por meio da aplicação do plano de verificação.

* RNF01: o sistema será prototipado no microcontrolador PIC18F4520;
* RNF02: o sistema será simulado na ferramenta de simulação Proteus;
* RNF03: o código da aplicação será escrito em linguagem c;
* RNF04: será utilizado o ambiente de desenvolvimento MPLAB;
* RNF05: o protótipo deverá ter uma vazão de pelo menos 10 frames/segundo (conforme a aplicação);
* RNF06: o sistema deverá ser exibido no display LM041L 16x4
* RNF07: as entradas de usuário serão por meio de um teclado.
* RNF08: o teclado de entrada deve possuir os seguintes botões: CIMA, BAIXO, ESQUERDA, DIREITA, ENTER e ESCAPE.
  + 1. Regras de negócio

Nesta seção, será apresentado as regras de negócio do sistema a ser desenvolvido.

* RN01: O menu principal deve ter três opções: jogar, pontuações e créditos;
* RN02: A pontuação deve ser incrementada a cada asteroide desviado;
* RN03: As pontuações devem estar em ordem descendente;
* RN04: Na ocorrência de uma colisão o jogo deve ser interrompido.
* RN05: Ao ser interrompido por uma colisão, o sistema deve informar a pontuação final.
* RN06: Ao ser interrompido por uma colisão e apresentar a pontuação final o sistema deve voltar ao menu principal.
* RN07: A nave poderá ser movimentada nos eixos XY.

**2 Referências**

# [SanticN4N](https://www.instructables.com/member/SanticN4N/), (2011). Arduino Asteroid Game.

https://www.instructables.com/id/Arduino-Asteroid-Game/