

PONG

Relatório de Laboratório de Computadores

Pedro Miguel Jesus da Silva (up201907523)

Miguel Azevedo Lopes (up201704590)

Grupo g04

Novembro 2020 - Janeiro 2021

Conteúdo

1	Instruções de uso	3
2	Estado do projeto	3
2.1	Timer	3
2.2	Teclado	3
2.3	Rato	4
2.4	Placa gráfica	4
2.5	RTC	4
3	Estrutura de código	5
3.1	Módulos	5
3.2	Timer	5
3.3	RTC	5
3.4	Keyboard	5
3.5	Mouse	5
3.6	Placa gráfica	5
3.7	Menu	5
3.8	Game sprites	5
3.9	Game	6
3.10	Peso relativo dos módulos	6
3.11	Código externo	6
3.12	Gráfico de chamada das funções	7
4	Detalhes de implementação	7
5	Conclusões	8

1 Instruções de uso

Para começar o jogo basta só executar "lcom run proj" na pasta src.

2 Estado do projeto

Periférico	Função	Interrupções
Timer	Ajuda no desenho de frames e a controlar eventos do jogo	sim
Teclado	Ajuda no movimento das barras	sim
Rato	Navegação dos menus	sim
Placa gráfica	Desenho	não possui
RTC	leitura da data para apresentação dos scores	não
UART	Não foi usado	não

2.1 Timer

As interrupções to timer foram usadas para não só contar o tempo entre frames para poder desenhar o jogo no ecrã, como também aumentar a velocidade da bola dado um certo número de ticks e contar o tempo de jogo que se apresenta no ecrã.

Funções principais:

- update_game_elapsed_time() - incrementa uma unidade do tempo de jogo
- reset_game_elapsed_time() - faz reset ao tempo de jogo quando alguém ganha
- timer_ih() - incrementa uma unidade no número de interrupções recebidas do timer
- timer_set_frequency() - muda a frequência de um dos times. Não foi usada no projeto

2.2 Teclado

O teclado foi usado para receber a input do jogador e movimentar as barras consoante isso.

Funções principais:

- keyboard_read_status() - lê o status do kbc;
- keyboard_read_buffer() - lê o scancode no buffer;
- keyboard_check_status() - processa o status do kbc
- kbc_ih() - lê o scancode no buffer quando é recebida uma interrupção se a informação for válida;
- keyboard_write_command() - não foi usado no projeto;
- update_state() - faz update no estado das barras;

2.3 Rato

O rato foi usado para receber a input do jogador e movimentar os menus.

Funções principais:

- mouse_handle_packet() - faz mutação de um packet com os bytes recebidos
- mouse_ih() - lê o scancode no buffer quando é recebida uma interrupção se a informação for válida;
- check_output_buffer() - verifica se o output buffer tem informação válida;
- mouse_update_state() - faz update do estado do cursor;

2.4 Placa gráfica

A placa gráfica foi usada para desenhar as letras, números e os elementos do jogo como as barras e a bola. O modo usado foi o indexado 0x105 dado que o jogo não exige cenários muito complexos. A colisão entre os elementos do jogo foi feita com o algoritmo simples de interseção de retângulos.

Funções principais:

- vg_init() - inicia em modo gráfico com o dado modo;
- map_vm() - mapeia a memória virtual;
- vg_colour_pixel() - dá cor a um dado pixel no ecrã;
- vg_draw_xpm() - desenha no ecrã uma imagem no formato xpm;
- move_xmp() - movimenta no ecrã uma imagem no formato xpm;
- move_ball() - movimenta a bola;
- reset_ball() - põe a bola no sitio quando é marcado um ponto;
- init_ball() - inicia a bola na posição;
- drawBars() - desenha as barras;

2.5 RTC

O real time clock foi usado para ler as datas e as horas para poder usar para apresentar os scores do jogo.

Funções principais:

- rtc_data_io() - permite escrever ou ler num dado registo;
- read_date() - lê todos os registos relevantes;
- bcd_conversion() - converte os valores de bcd para decimal;
- rtc_ih() - indica a fonte da interrupção do RTC;
- rtc_enable_interrupts() faz set aos bits do registo B para fazer enable das interrupções;

3 Estrutura de código

3.1 Módulos

3.2 Timer

Igual ao lab 2.

Participação: Pedro 80%, Miguel 20%

3.3 RTC

Participação: Pedro 75%, Miguel 25%

3.4 Keyboard

A diferença em relação às funções do lab é que tivemos de gerir a possibilidade de mais de uma tecla ser premida ao mesmo tempo, visto que os jogadores jogam no mesmo teclado.

Participação: Pedro 50%, Miguel 50%

3.5 Mouse

As funções são as mesmas que desenvolvemos no lab, sendo que agora utilizamos o rato com uma representação gráfica, o cursor.

Participação: Pedro 20%, Miguel 80%

3.6 Placa gráfica

Igual ao lab 5.

Participação: Pedro 50%, Miguel 50%

3.7 Menu

É neste módulo que se inicializam os sprites dos menus(menu principal, menu de pausa e os high scores). Inclui também funções responsáveis pela utilização do rato como forma de seleção de opções no menu.

Participação: Pedro 50%, Miguel 50%

3.8 Game sprites

Consiste maioritariamente em inicializar variáveis de posição e pointers com os sprites do jogo. Contem também algumas funções responsáveis pelo movimento das barras e da bola, bem como deteção de colisão da bola com outros elementos e finalmente uma função que no fim do jogo liberta a memória alocada aos pointers.

Participação: Pedro 25%, Miguel 75%

3.9 Game

Contém o loop que é responsável pelo jogo, que inclui o driver receive. Para além disso contém também a função responsável pela gestão dos pontos, bem como as funções para ler/escrever o ficheiro que contém os high scores.

Participação: Pedro 50%, Miguel 50%

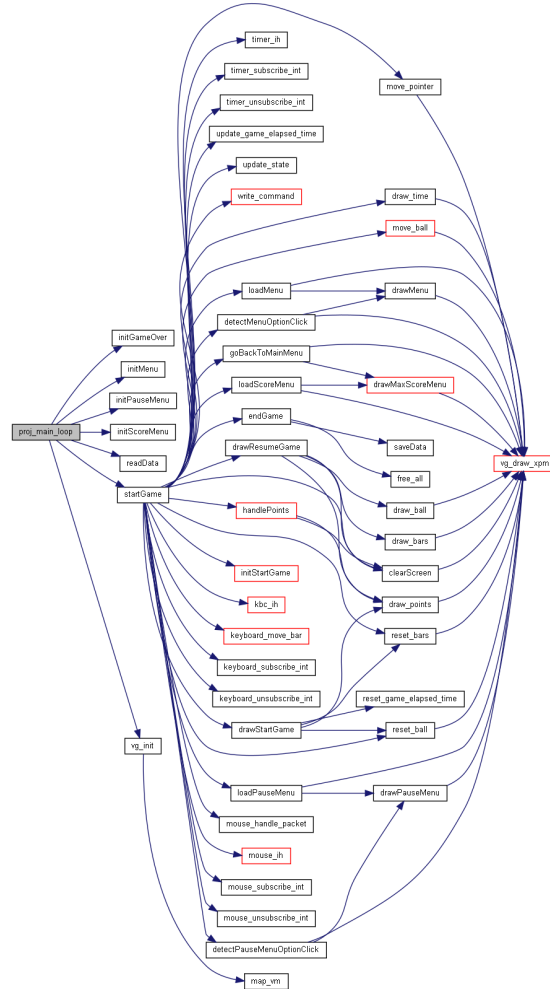
3.10 Peso relativo dos módulos

Módulo	Percentagem
Timer	5
keyboard	6
mouse	6
video card	10
RTC	6
menu	23
game sprites	22
game	22

3.11 Código externo

Não houve necessidade de usar código externo.

3.12 Gráfico de chamada das funções



4 Detalhes de implementação

Não foram usadas as interrupções do RTC mas está presente o código para o seu uso. Converteu-se os valores em bcd usando uma fórmula em vez de alterar o bit que punha os valores em binário. A data e a hora foram lidas com cuidado esperando sempre que não esteja a ocorrer nenhum update dos registos. Reparou-se que o minix na virtualbox produzia mal o resultado do dia da semana - no nosso caso lia o dia da semana com um dia de atraso e pesquisou-se na net o problema e viu-se que era um bug inerente do registo do dia da semana no minix.

5 Conclusões

Na sequência da realização deste trabalho enfrentamos várias dificuldades, nomeadamente a falta de documentação atualizada (algumas das páginas referenciadas na página da disciplina não estão online) e sugerimos que de futuro esta informação fosse sintetizada e compilada apenas num documento. Para além disso, surgiram alguns problemas na implementação gráfica (aparecimento de lag quando a utilização da parte gráfica é mais intensa), que ao fim de muito tempo a tentar diagnosticar, sem sucesso, atribuímos ao mal funcionamento da plataforma utilizada para correr os programas. Dito isto, ambos nós consideramos a cadeira de LCOM interessante e que contribuiu bastante para o nosso desenvolvimento enquanto estudantes e futuros profissionais.