Semester work(Game Pong)

Topic

The theme of our semester work is **Pong**

Purpose

Write a Pong game and play it on MZAPO

Name of participants:

Daniel Koval, Victor Remel

Platforms used:

Github: https://github.com/remelvic/APO

User Manual

How to play

The rules are the same as in the normal game, the controls are as follows: about the left stick: **w** - up, **s** - down. About the right stick: **↑(8)** - up, **↓(2)** - down. **n** - new game, **q** - exit.

How to launch

You have to compile the code using the Makefile in the terminal.

- *make*

In the Makefile, change the IP address of the MZAPO to which you want to connect.

- *TARGET_IP ?= 192.168.XXX.XXX*

Write the command make run and run the Pong.c file with our game.

- *make run*

For the Developer

Function void draw_on_screen(unsigned char *parlcd_mem_base)

- *Lets us draw on the screen*
- **Parameter** *parlcd_mem_base*

Function void init_termios()

- *Always reads from stdin waits for w or s or ↑(8) or ↓(2)*

Function unsigned int hsv2rgb_lcd(int hue, int saturation, int color)

- *This function works with the color of our elements (sticks and ball)*

- ~~**Parameter** *hue - always 255*~~
- ~~**Parameter** *saturation - always 255*~~
- **Parameter** *color*
- **Return** *color*

Function void draw_player_score(unsigned short color)

- *This function writes out the goals scored counter*
- **Parameter** *color - color counter*

Function void draw_pixel(int x, int y, unsigned color)

- *The function gives the pixels a color, thereby writing them out (they appear on the screen)*
- **Parameter** *x - width*
- **Parameter** *y - height*
- **Parameter** *color*

Function void draw_pixel_big(int x, int y, unsigned short color)

- *The function "enlarges" the pixel so that it is visible on the screen*
- **Parameter** *x - width*
- **Parameter** *y - height*
- **Parameter** *color*

Function void draw_center_stick_and_borders(unsigned char *parlcd_mem_base)

- *The function will draw a center line and sticks on the left and right side*
- ~~**Parameter** *parlcd_mem_base*~~

Function void draw_ball(ball_t ball)

- *The function draws a ball on the screen*
- **Parameter** *ball - we transfer the speed of the ball, its size and position*

Function _Bool move_ball(ball_t *ball, stick_t player1, stick_t player2)

- *The function processes the movement of the ball in all directions*
- **Parameter** *ball - we transfer the speed of the ball, its size and position*
- **Parameter** *player1 - we pass the dimensions and position of the first stick*
- **Parameter** *player2 - we pass the dimensions and position of the second stick*
- **Return** *- if the ball moves left or right, then the function will return true; otherwise, it will change the position of the ball on the screen and return false*

Function void draw_stick(stick_t stick)

- *The function draws our sticks on the screen*

- **Parameter** *stick - passing the dimensions and parameters of the stick*

Function void move_stick(stick_t *stick, int direction)

- *The function will moves our sticks*
- **Parameter** *stick - in this case, we get the coordinates of the stick*
- **Parameter** *direction - where will we move our stick*

Function void draw_char(int x, int y, font_descriptor_t *fdes, char ch)

- *The function draws the character ch at x and y coordinates*
- **Parameter** *x* - *width*
- **Parameter** *y* - *height*
- **Parameter** *fdes - representation of our symbol*
- **Parameter** *ch - our symbol that we want to draw*

Function void draw_text(char *str, int x, int y, font_descriptor_t *fdes)

- *The function writes out the whole text*
- **Parameter** *str - our line that we want to write out*
- **Parameter** *x* - *width*
- **Parameter** *y* - *height*
- **Parameter** *fdes* - *"storage" of representations of our symbols*

Function int char_width(font_descriptor_t *fdes, int ch)

- *The function determines the width of the selected character based on the "stock" of representations fdes*
- **Parameter** *fdes* - *"storage" of representations of our symbols*
- **Parameter** *ch - character*
- **Return** *the width of our character*

Function void clear_map(int ptr, unsigned char *parlcd_mem_base)

- *The function clears the screen after finishing the game (After pressing the q button)*
- **Par~~ameter~~** ~~*parlcd_mem_base*~~

Function int main(int argc, char *argv[])

- *The main function that launches our game*