
Project: Simulation Pong Game

Real-Time System

Adriano Oliveira (67851) Rui Oliveira (68779)

Teacher: Paulo Pedreiras

19/12/2016

Goal/Problem

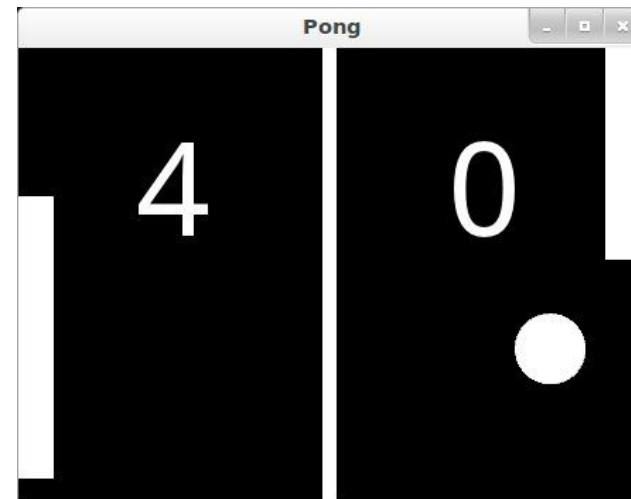
- **Simulation of a game using an RTOS**
- **Popular pong game**
- **Technologies used**
 - Language C
 - Xenomai 2.6
 - SDL 1.2 (user interface)
- **Main components**
 - Paddle aka bars
 - Ball



Specific aspects - implementation (1)

- **ball struct**

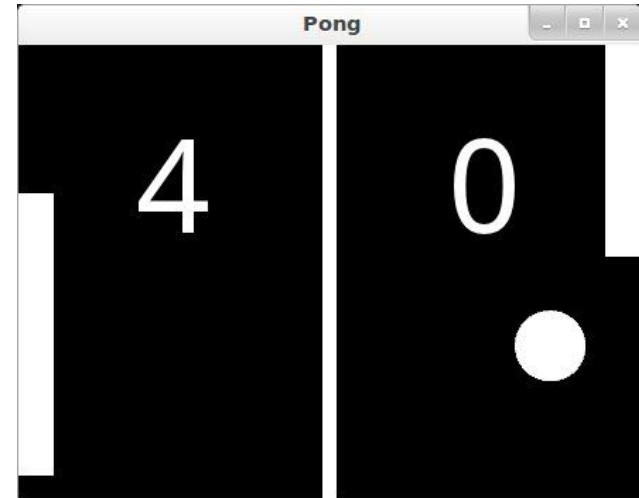
```
typedef struct ball_s {  
    int x, y; /* position on the screen */  
    int w,h; /* ball width and height */  
    int dx, dy; /* movement vector */  
} ball_t;
```



Specific aspects - implementation (2)

- paddle struct

```
typedef struct paddle {  
    int x,y; /* bar position in the window*/  
    int w,h; /* Dimensions of the bars*/  
} paddle_t;  
  
static paddle_t paddle[2]  
/* paddle righth and left*/
```



Specific aspects - implementation (3)

- **Tasks created/used in xenomai**
 - **Cinematics AKA god**
 - Compute next position of the ball - with collision checking
 - **Move paddle User $\uparrow \downarrow$**
 - Check pressed key to move paddle - change coordinate **y** (two or one user)
 - **Move paddle Artificial Intelligence (created only when playing with computer)**
 - Moves in **y** according to **y** coordinate of the ball

Specific aspects - implementation (4)

- **Tasks created/used in xenomai**

- **Cinematics AKA god**
 - `#define TASK_CINEMATICA_PERIOD_NS 100000000 // Task period, in ns`
- **Move paddle User $\uparrow \downarrow$**
 - `define TASK_MOVE_PADDLE_PERIOD_NS 99000000 // Task period, in ns`
- **Move paddle Artificial Intelligence (created only when playing with computer)**
 - `#define TASK_MOVE_PADDLE_AI_PERIOD_NS 99000000 // Task period, in ns`

- **Priority of tasks**

- same in all

Results

Cinematics (ns)	Move paddle User (ns)	Move paddle AI (ns)
100035841	95029672	94925595
99789062	85763888	86647424
99786590	92794659	93391041

- **Period Cinematics > Period move paddle \approx Period move paddle AI**

Execution + Demo

- **./pong**
 - multiplayer (player 1: keyUP, keyDown; player 2: W, S)
- **./pong debug**
 - show ball position and keys pressed
- **./pong computer**
 - Play against computer
- **./pong computer debug**
 - Two lastest features

