

UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY

Retro videohra PONG
Základy elektroniky
Projekt

Úvod

Videohra PONG je jedna z najstarších na svete. Vydala ju spoločnosť ATARI v roku 1972. Jedná sa o dvojdimenzionálnu hru na štýl tenisu, kde na ľavej a pravej strane obrazovky sú hráčom ovládané platformy, ktoré odrážajú loptičku. Víťazom je ten, ktorého oponent nedokáže odraziť loptičku.



Obrázok č. 1 Hra PONG

Na obrázku č. 1 možno vidieť hru PONG. V strede je predeľovacia čiara, a vľavo a vpravo hore je ukazovateľ aktuálneho skóre.

V mojom projekte budem túto hru simulovať na LED displeji, pomocou mikrokontroléra Arduino.

Použité komponenty

1x Arduino Uno

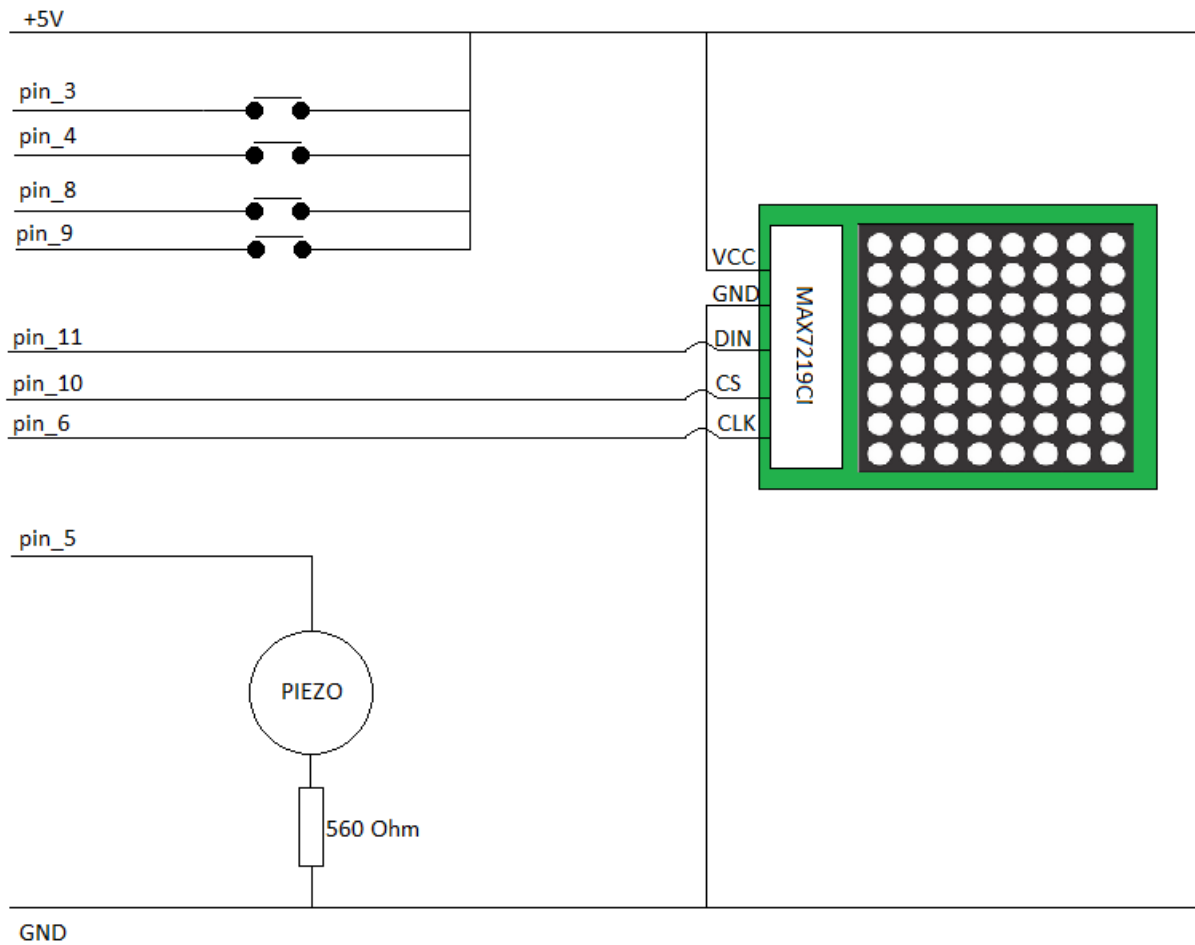
1x Maticový LED display 8x8 s MAX7219

1x Membránová klávesnica 1x4

1x 512 Ohm rezistor

1x Piezo bzučiak

Schéma zapojenia



Obrázok č. 2 Schéma zapojenia

Implementácia

Celú hru si vieme predstaviť ako jeden nekonečný cyklus, kde jedna iterácia týmto cyklom predstavuje tzv. "tick".

```
void loop() {  
    move_player(); // pohni hracom  
    move_ai(2); // pohni pocitacom  
    collision_detection(); // pohni loptickou a skontroluj koliziu  
    check_result(); // skontroluj stav hry  
    delay(delaytime); } // uspi sa na nejaky cas
```

Obrázok č. 3 Hlavná slučka programu

Funkcia `move_player()` kontroluje či sa hráči chcú pohnúť. Robí to tak že skontroluje či je daný gombík stlačený, tak že skontroluje napätie na danom pine.

Funkcia `move_ai(2)` pohne počítačom. Umelá inteligencia je veľmi primitívna, ale efektívna. Počítač len skontroluje či y-nová pozícia jeho padla je rovnaká ako y-nová pozícia + zmena pozície loptičky. Keď nie je, tak sa posunie smerom k nej (či už hore alebo dole).

Funkcia `collision_detection` hybe s loptičkou. Loptička má uloženú svoju x a y pozíciu, a zmenu x a y (`dx` a `dy`).

```
// game over, lopticka je na urovni padla
    if (ball_posX + ball_dX < 0 || ball_posX + ball_dX > 7){
        game_over = 1;
        return;
    }

// odrazi sa o horny alebo dolny bok
    if (ball_posY + ball_dY < 0 || ball_posY + ball_dY > 7){
        ball_dY = -ball_dY;
    }

// player 1 paddle
    if (ball_posX + ball_dX == player1_posX && (ball_posY + ball_dY == player1_posY || ball_posY +
ball_dY == player1_posY + 1)){
        ball_dX = -ball_dX;
    }

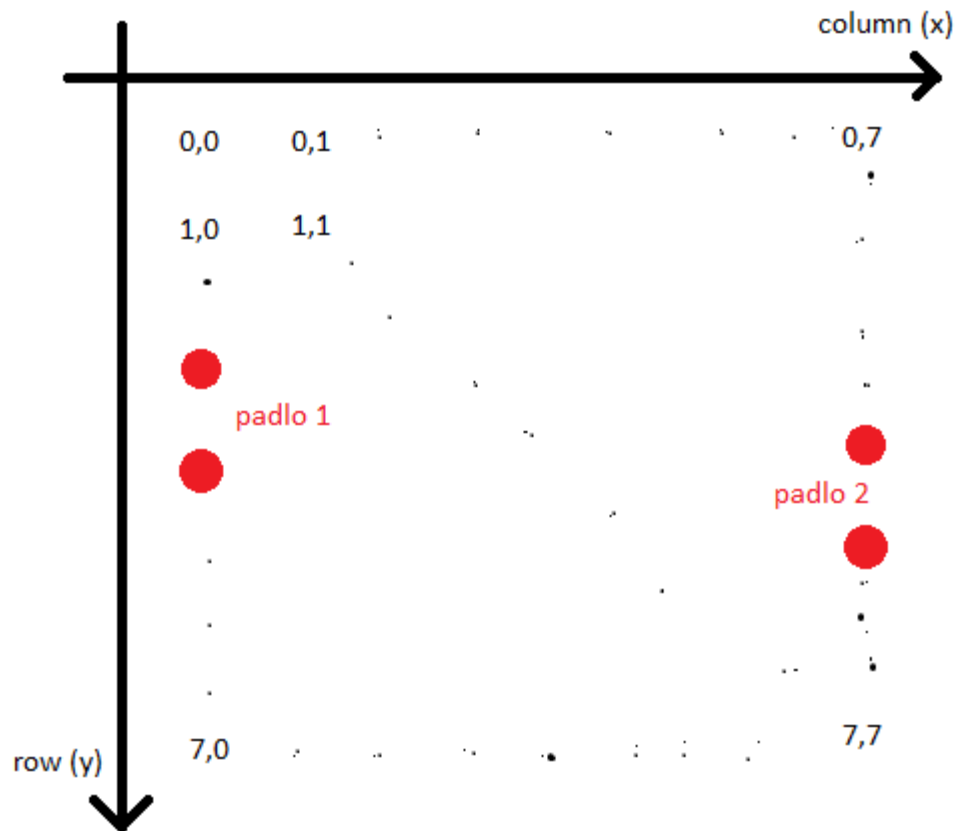
// player 2 paddle
    if (ball_posX + ball_dX == player2_posX && (ball_posY + ball_dY == player2_posY || ball_posY +
ball_dY == player2_posY + 1)){
        ball_dX = -ball_dX;
    }
```

Obrázok č. 4 Logika loptičky

Funkcia `check_result()` len skontroluje stav premennej `game_over`. V prípade že je `true`, ukončí hru a zapne novú.

Vykresľovanie ťahu

Ledky sú usporiadané v matici takto.



Obrázok č. 5 Abstrakcia LED displeja

Na zjednodušenie vykresľovania na LED displej som použila LED display driver MAX7219 a k nemu knižnicu `LedControl`. Táto knižnica obsahuje funkciu

```
void setLed(int addr, int row, int col, boolean state);
```

s ktorou môžem jednoducho vypnúť/zapnúť (`state`) LEDku na pozícií [`row,column`]. Túto funkciu som využila pri akomkoľvek pohybe na obrazovke.

```
lc.setLed(0,ball_posY,ball_posX,false); // vypni starú pozíciu  
ball_posX += ball_dX;  
ball_posY += ball_dY;  
lc.setLed(0,ball_posY,ball_posX,true); // zapni novú pozíciu
```

Obrázok č. 6 Vykreslenie pohybu loptičky

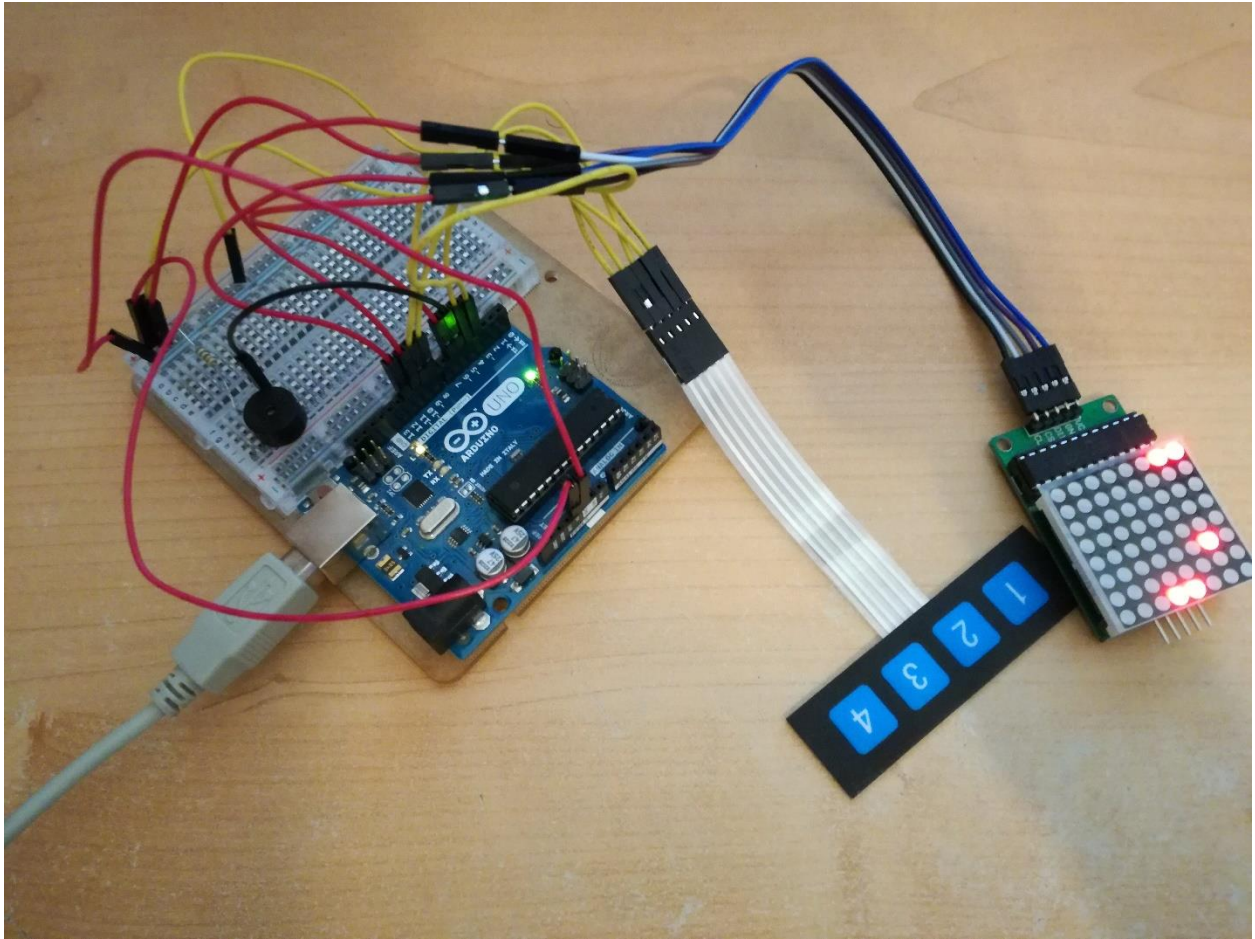
Bzučiak

Pri každej kolízii loptičky (stena, padlo), zabučí bzučiak pre efekt. O tento bzukot sa stará funkcia buzz().

```
void buzz(){  
    tone(buzzerPin, 2000); // nastav ton bzuciaku  
    delay(50); // bzuc 50ms  
    noTone(buzzerPin); // vypni bzuciak  
}
```

Obrázok č. 7 Bzučiak

Obrázok zapojenia



Obrázok č. 8 Fotografia zapojenia

Zdroje

LedControl knižnica <https://github.com/wayoda/LedControl/blob/master/src/LedControl.h>