**2190151 Computer Programming Lab**

Lab 6: King Pong Game

Objectives:

1. Student be able to use potentiometer in their IoT project

2. Student be able to write simple game on M5Stack

**Background Theory**

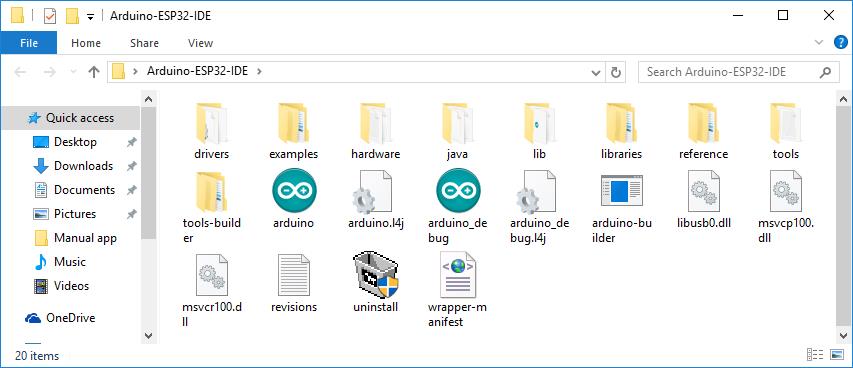
Table tennis (sometimes colloquially mistakenly used the name ping-pong) — Olympic sport, athletic game ball, which uses a racket and a table separated by a net. The game can take place between two opponents or two pairs of opponents. The task of the players is to use the rackets to send the ball to the opponent so that he could not return it back in accordance with the rules. p.s. In our case, the player will play with himself.

More on Wiki: <https://en.wikipedia.org/wiki/Table_tennis>

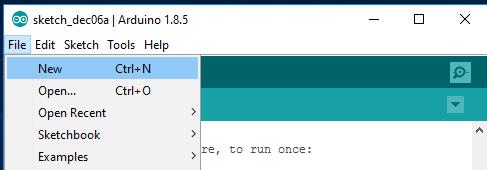
**List of components for the lesson**

* PC/MAC;
* M5STACK CORE;
* USB-C cable from standard set;
* variable resistor;
* colored wires from the standard set (type plug-socket);
* colored wires not from the standard set (socket type);
* wire cutters;
* shrinkage;
* needle;
* soldering iron and solder.

**Task1: Basic pong game**  
**Step 1. Go to the folder Arduino-ESP32-IDE and open the app arduino.exe (Fig. 1);**

  
Figure 1. The archive with the Arduino IDE extracted to the folder

**Step 2. Select New in the File menu (Fig. 1.1);**

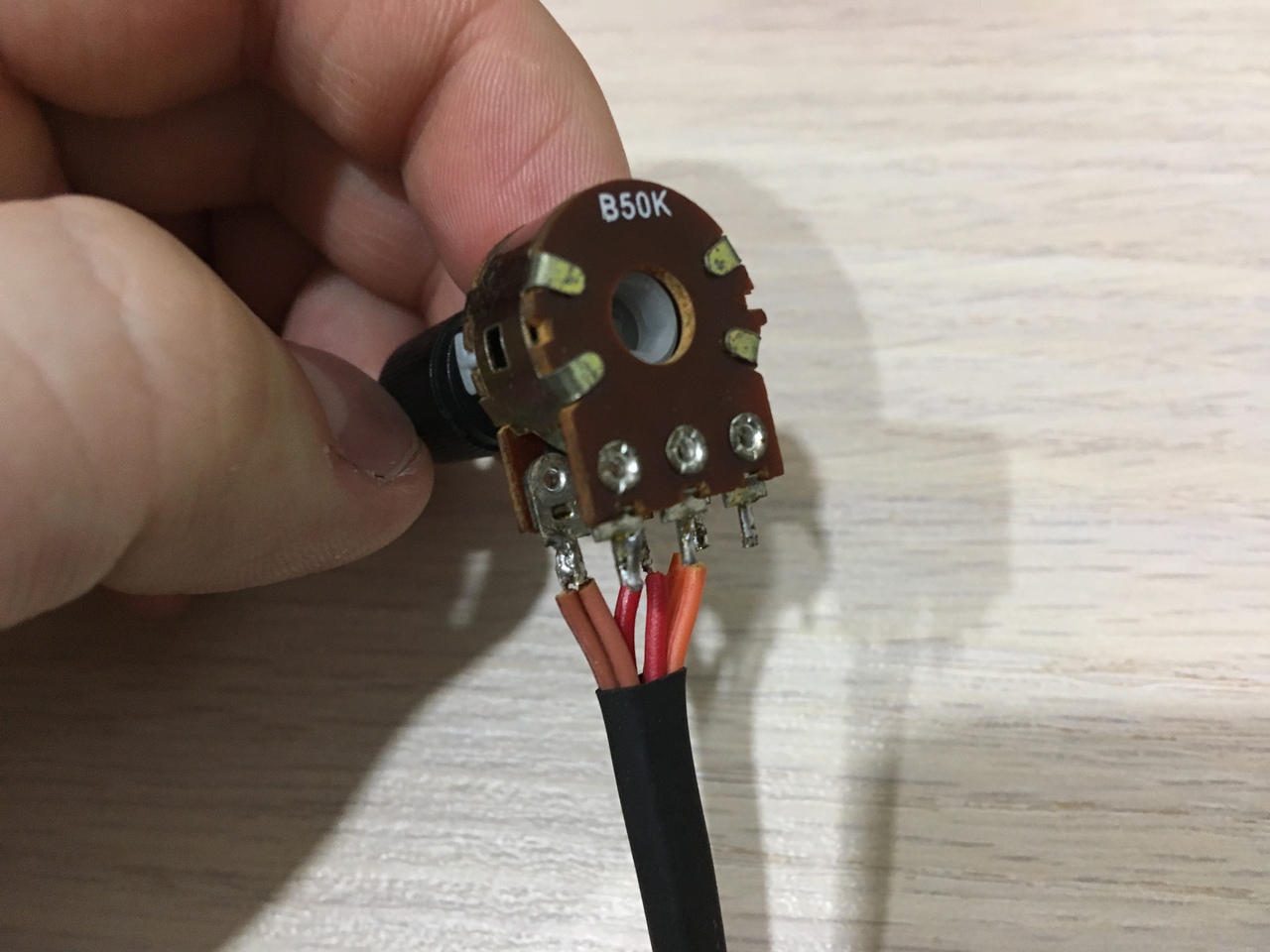
  
Figure 1.1. Create a new sketch

**Step 3. Select Include Library, M5Stack in the Sketch menu (Fig. 1.4);**

**Step 4. Downloads**

**Sketch + LED library (GitHub):** [**https://github.com/dsiberia9s/Lesson-20\_KingPong\_Game**](https://github.com/dsiberia9s/Lesson-20_KingPong_Game)

**Step 5: connect potentiometer to M5stack**

**Example sketch (Fig. 3);**

#include <M5Stack.h>

~~#include "esp32\_digital\_led\_lib.h"~~

~~strand\_t m\_sLeds = {.rmtChannel = 0, .gpioNum = 15, .ledType = LED\_WS2812B\_V3, .brightLimit = 32, .numPixels = 10, .pixels = nullptr, .\_stateVars = nullptr};~~

extern unsigned char logo[];

int screen\_width = 320;

int screen\_height = 240;

int raket\_position;

long interval;

int ball\_x;

int ball\_y;

int ball\_r;

int accel\_x;

int accel\_y;

int score;

~~void ledBar(int R, int G, int B, int M) {~~

~~if ((M < 0) || (M > 12)) return;~~

~~if (M == 11) // right~~

~~{~~

~~for (int i = 0; i < 5; i++)~~

~~{~~

~~m\_sLeds.pixels[i] = pixelFromRGBW(R, G, B, 0);~~

~~}~~

~~}~~

~~else if (M == 10) // left~~

~~{~~

~~for (int i = 5; i < 10; i++)~~

~~{~~

~~m\_sLeds.pixels[i] = pixelFromRGBW(R, G, B, 0);~~

~~}~~

~~}~~

~~else if (M == 12) // all~~

~~{~~

~~for (int i = 0; i < 10; i++)~~

~~{~~

~~m\_sLeds.pixels[i] = pixelFromRGBW(R, G, B, 0);~~

~~}~~

~~}~~

~~else~~

~~{~~

~~m\_sLeds.pixels[M] = pixelFromRGBW(R, G, B, 0);~~

~~}~~

~~digitalLeds\_updatePixels(&m\_sLeds);~~

~~}~~

void start() {

M5.Lcd.fillScreen(0x0000);

interval = 60;

ball\_x = 160;

ball\_y = 120;

ball\_r = 6;

accel\_x = 5;

accel\_y = 5;

score = 0;

M5.Lcd.drawBitmap(0, 0, screen\_width, screen\_height, (uint16\_t \*)logo);

M5.Lcd.setTextSize(2);

long previousMillis = 0;

long interval = 500;

int color = 0x0000;

while(true)

{

unsigned long currentMillis = millis();

if (currentMillis - previousMillis > interval) {

previousMillis = currentMillis;

color = (color == 0x7bef) ? 0xffff : 0x7bef;

M5.Lcd.setCursor(35, 200);

M5.Lcd.setTextColor(color);

M5.Lcd.print("press any key to play");

}

M5.update();

if (M5.BtnA.wasPressed() || M5.BtnB.wasPressed() || M5.BtnC.wasPressed()) break;

}

M5.Lcd.fillScreen(0x0000);

M5.Speaker.tone(800);

delay(100);

M5.Speaker.tone(1200);

delay(100);

M5.Speaker.mute();

delay(500);

M5.Lcd.fillScreen(0xffff);

}

void game() {

M5.Lcd.fillCircle(ball\_x, ball\_y, ball\_r, 0x7bef);

long previousMillis = 0;

while (true) {

int voltage = analogRead(36) \* 3400 / 4096;

int percentage = voltage \* 100 / 3400;

raket\_position = map(percentage, 0, 100, 0, 10);

int raket\_width = 20;

int raket\_height = 40;

int raket\_margin = 10;

int x, y, color, raket\_x, raket\_y;

for (int i = 0; i < 10; i++){

if (i < 5){

x = 0;

y = i \* (raket\_height + raket\_margin);

}

else {

x = screen\_width - raket\_width;

y = (9 - i) \* (raket\_height + raket\_margin);

}

color = (i == raket\_position) ? RED : WHITE;

M5.Lcd.fillRect(x, y, raket\_width, raket\_height, color);

~~ledBar(0, 0, 0, 12);~~

~~ledBar(255, 0, 0, 9 - raket\_position);~~

if (i == raket\_position){

raket\_x = x;

raket\_y = y;

}

}

unsigned long currentMillis = millis();

if (currentMillis - previousMillis > interval) {

previousMillis = currentMillis;

M5.Lcd.fillCircle(ball\_x, ball\_y, ball\_r, WHITE);

ball\_x += accel\_x;

ball\_y += accel\_y;

M5.Lcd.fillCircle(ball\_x, ball\_y, ball\_r, 0x7bef);

if (ball\_y <= 0)

accel\_y \*= -1;

if (ball\_y >= 240)

accel\_y \*= -1;

if (raket\_position > 4){

if ((ball\_x + ball\_r >= raket\_x) && (ball\_y + ball\_r >= raket\_y) && (ball\_y - ball\_r <= raket\_y + raket\_height)){

accel\_x \*= -1;

accel\_y \*= 1;

repulse();

}

}

else{

if ((ball\_x - ball\_r <= raket\_x + raket\_width) && (ball\_y + ball\_r >= raket\_y) && (ball\_y - ball\_r <= raket\_y + raket\_height)) {

accel\_x \*= -1;

accel\_y \*= 1;

repulse();

}

}

if ((ball\_x < 0) || (ball\_x > screen\_width) || (ball\_y < 0) || (ball\_y > screen\_height)){

gameover();

return;

}

}

}

}

void repulse() {

~~ledBar(0, 0, 0, 12);~~

~~ledBar(0, 255, 0, 12);~~

M5.Speaker.tone(1800);

delay(50);

M5.Speaker.mute();

~~ledBar(255, 0, 0, 9 - raket\_position);~~

score++;

interval -= (interval >= 25) ? 5 : 0;

}

void gameover() {

M5.Lcd.fillScreen(0x0000);

M5.Speaker.tone(1200);

delay(100);

M5.Speaker.tone(1000);

delay(100);

M5.Speaker.tone(800);

delay(100);

M5.Speaker.tone(1100);

delay(100);

M5.Speaker.mute();

M5.Lcd.setCursor(100, 100);

M5.Lcd.setTextColor(0xffff);

M5.Lcd.print("GAME OVER");

delay(1000);

~~ledBar(0, 0, 0, 12);~~

for (int i = 0; i < score; i++)

{

M5.Lcd.setCursor(100, 120);

M5.Lcd.setTextColor(0xffff);

M5.Lcd.print("SCORE: ");

M5.Lcd.setTextColor(0x0000);

M5.Lcd.print((i < 1) ? 0 : i);

M5.Lcd.setCursor(100, 120);

M5.Lcd.setTextColor(0xffff);

M5.Lcd.print("SCORE: ");

M5.Lcd.print(i + 1);

~~ledBar(random(0, 256), random(0, 256), random(0, 256), 12);~~

M5.Speaker.tone(1200);

delay(100);

M5.Speaker.mute();

}

~~ledBar(0, 0, 0, 12);~~

delay(2000);

}

void setup(){

M5.begin();

pinMode(36, INPUT);

~~pinMode(15, OUTPUT);~~

~~digitalWrite (15, LOW);~~

~~if (digitalLeds\_initStrands(&m\_sLeds, 1)) {~~

~~Serial.println("Can't init LED driver().");~~

~~}~~

~~digitalLeds\_resetPixels(&m\_sLeds);~~

}

void loop() {

start();

game();

}

**Section\_\_\_\_\_\_\_\_\_\_\_\_\_date\_\_\_\_\_\_\_\_\_\_**

**Name\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Student ID\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Laboratory 6 Pong Game:**

**Task1: One player**

Graded by \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Time\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Task 2: Two players**

Graded by \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Time\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**What do you learn from this lab?**