

Proyecto Ultra Space

-¿De qué se trata el juego?

Se controla la bala moviendo la mano hacia arriba y hacia abajo por la proximidad ultrasónica del sensor.

Está basado en el juego flappy bird pero con algunas innovaciones como por ejemplo el poder disparar a los enemigos que se van acercando y el aumento de velocidad con el tiempo.

-Objetivo:

Hacer la mayor cantidad de puntos posibles y sobrevivir el mayor tiempo que pueda el jugador.

Dispararle a los enemigos que se van acercando al jugador.

-Pasos a seguir para realizarlo:

Paso 1: Reunir las piezas

Para este proyecto, necesitará 3 partes y 2 softwares descargados.

1. Una placa Arduino Uno
2. Sensor de ultrasonido.
3. Cables de puente macho a hembra.
4. Programa Arduino.
5. Processing.

Paso 2: Hacer las conexiones de Arduino UNO

En el sensor de ultrasonido, debería ver 4 pines. VCC (alimentación), Trig (para enviar los pings), Echo (de salida) y tierra.

1. Conecte el pin de alimentación al Arduino.
2. Conecte el Trig al pin 3 de entrada/salida digital del Arduino.
3. El de salida a Arduino Digital I/O Pin 2.
4. GND a cualquier pin GND disponible en el Arduino.

Paso 3: Programa la placa de Arduino

Antes de programar la placa, conectamos el Arduino a la computadora y abrimos Arduino Software. Una vez cargado, abra Serial Monitor y verificamos el número de puerto COM.

Alternativamente, se puede verificar en Herramientas -> Puerto. Debería poder ver el puerto COM. Usaremos este número para recopilar información desde el sensor hasta el software Processing. En nuestro caso, es COM 3.

Paso 4: Programar el juego

Explicación del código de Arduino:

const int trigPin=11; *//Puede usarse como cualquier otra variable pero su valor no puede ser cambiado en este caso 11*

int echoPin=10;

int safezone=10;

int boton;

void setup()

{

//inicializamos los dos pines del ultrasonidos

pinMode(trigPin,OUTPUT);

pinMode(echoPin,INPUT);

pinMode(2,INPUT);

Serial.begin(9600);

}

void loop(){

boton=digitalRead(2);

if (boton==HIGH){

Serial.write('a');

}

delay (100);

long duration,cm; *//MIDE LA DISTANCIA*

digitalWrite(trigPin,LOW); *//GENERA UN PULSO LIMPIO*

delayMicroseconds(2); *//ESPERA 2 MICROSEG*

digitalWrite(trigPin,HIGH); *//GENERA EL DISPARO*

delayMicroseconds(5); *//ESPERA 5 MICROSEG HASTA QUE SE ENVÍE EL DISPARO DEL PULSO*

digitalWrite(trigPin,LOW); *//VUELVE A GENERAR PULSO*

duration=pulseIn(echoPin,HIGH); *//MIDE EL TIEMPO ENTRE PULSOS EN MICROSEGUNDOS*

cm=duration*0.034; *//FORMULA PARA CONVERTIR DISTANCIA EN CM*

long inch= cm/2.54;

Serial.println(cm);

}

long microsecondsToCentimeters(long microseconds)

{

return microseconds/29/2; *//EL PULSO VIAJA DEL SENSOR Y VUELVE DE NUEVO POR LO QUE LO DIVIDIMOS POR 2 PARA TOMAR SOLO LA MITAD DE LA DISTANCIA TOTAL*

}

Luego en Processing busque esta línea:

myPort = new Serial(this, Serial.list()[0], 9600);

Tendrías que cambiar el puerto serie "Serial.list()[0]" -> el número "0" en código para el equivalente al puerto COM que está utilizando.

Puede usar el archivo adjunto "Serial.pde" para identificar

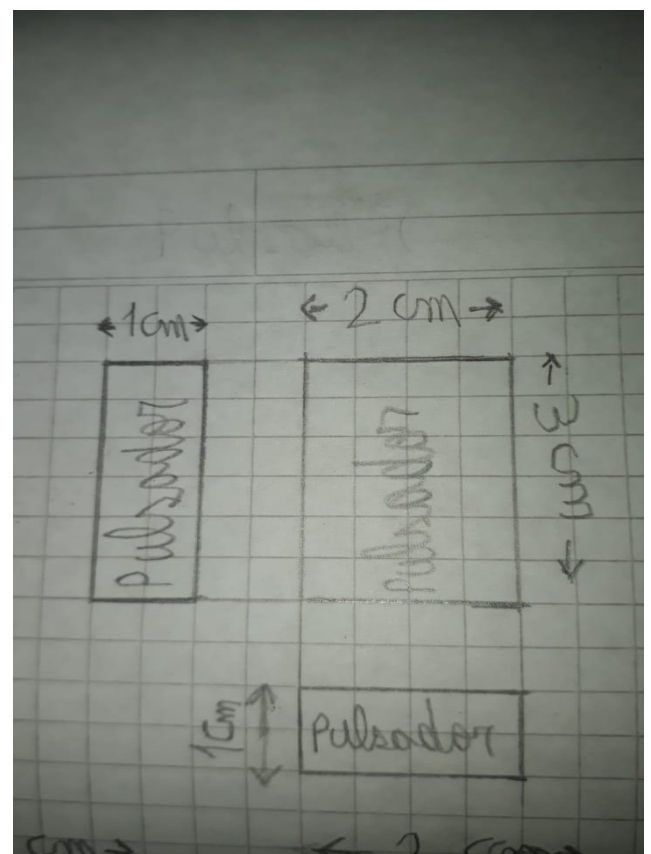
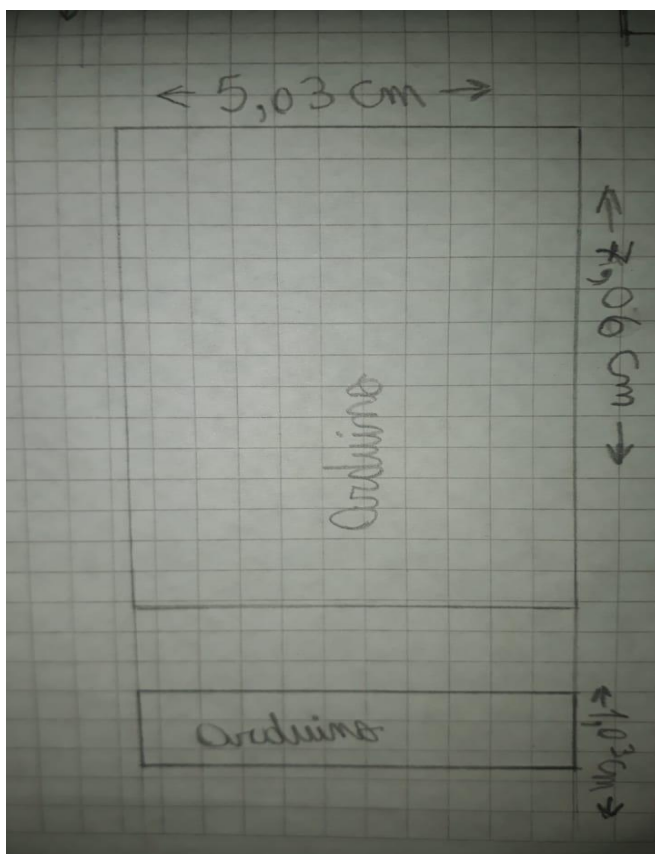
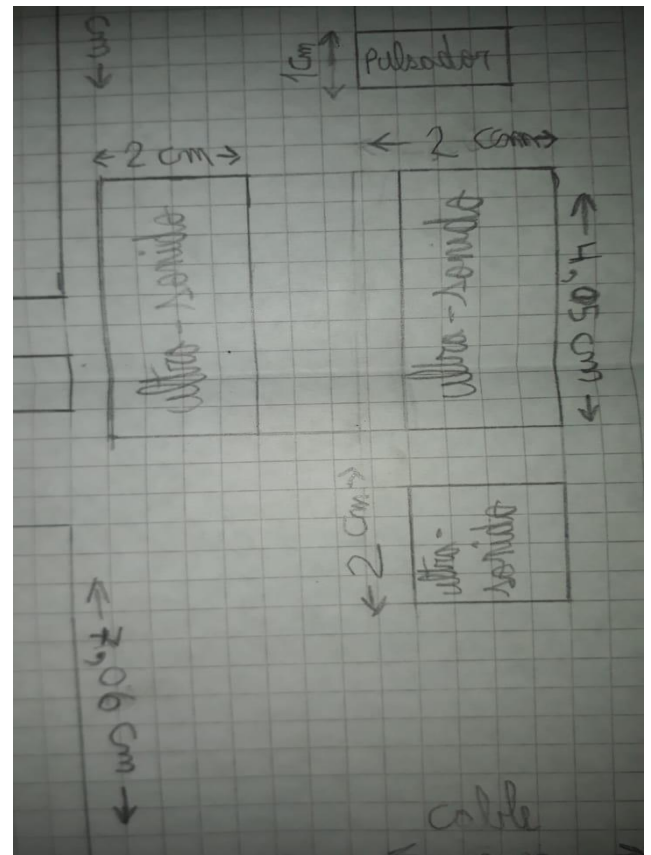
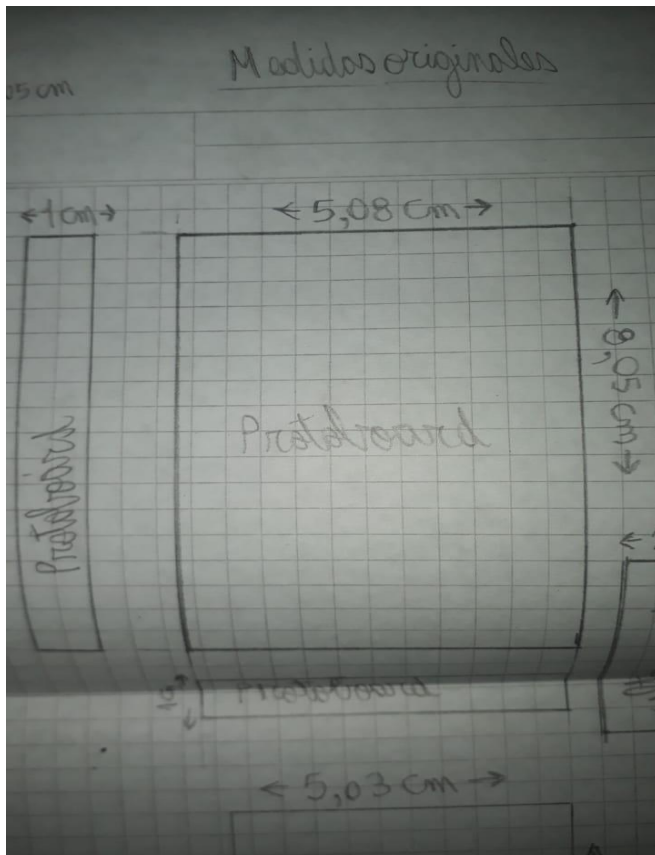
el puerto serie/puerto de procesamiento equivalente al COM Puerto.

La razón por la que estamos haciendo esto es que Processing el software no funciona directamente con el puerto COM.

El software de Arduino sólo enviará los datos si el sensor detecta un objeto que se mueve sobre él (en este caso de la mano del jugador.). Una vez que recibimos estos datos en el boceto de procesamiento, usaremos los valores para verificar el rango y hacer que actúe de acuerdo con el código.

Paso 5: Crear el arma para utilizarlo como mando

En este caso nosotros usamos cartón e hicimos el diseño en base a las medidas que teníamos de los materiales, acá están para que se puedan guiar y hacerlo a su manera.



Controles:

El personaje (La bala) se maneja con el sensor de ultrasonido acercando la mano o alejando, dependiendo de lo que haga el personaje sube o baja. Esto también se puede hacer pulsando las teclas "W" o "S".

Con el botón espacio (barra espaciadora del teclado) el jugador puede disparar a los enemigos.

Extras:

Intentamos agregar un botón al arduino para la acción de disparo pero no fue posible, el código está incompleto dentro de processing.

El juego puede contener varios bugs y a veces no funcionar sin razón.

