



# Instructivo CE Experience 2018

Ing. Computadores - TEC





ASEIC

# Introducción

Utilizando Processing se creará una imitación del juego Space Invaders. Este se va a comunicar por medio de un puerto serial con Arduino. Con Arduino se va crear un control para cada nave en el juego.

Para esto se ocupan:

- 6 botones
- 2 LED
- 2 Buzzer
- 1 Arduino UNO
- Jumpers
- 2 Protoboard



ASEIC

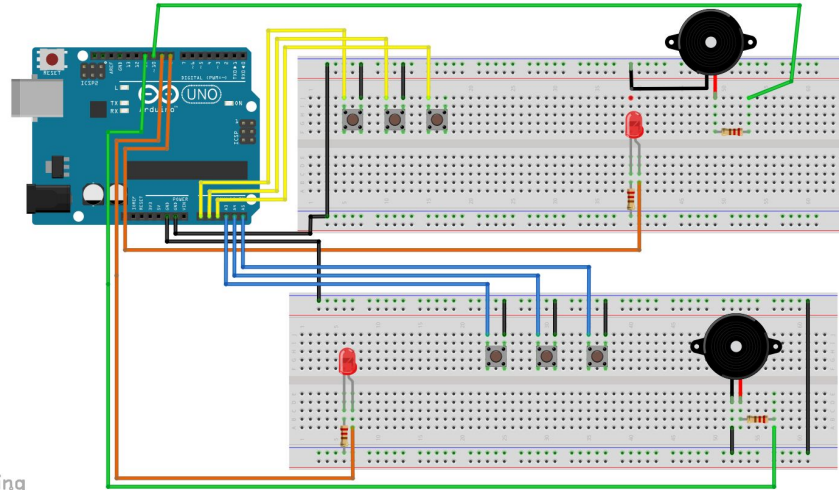
# Esquema

Se deben conectar los componentes como se muestra en la siguiente imagen.

Se deben respetar los pines en los cuales se conectan ya que así están programados.

La protoboard de arriba será el jugador 1 y el de abajo jugador 2.

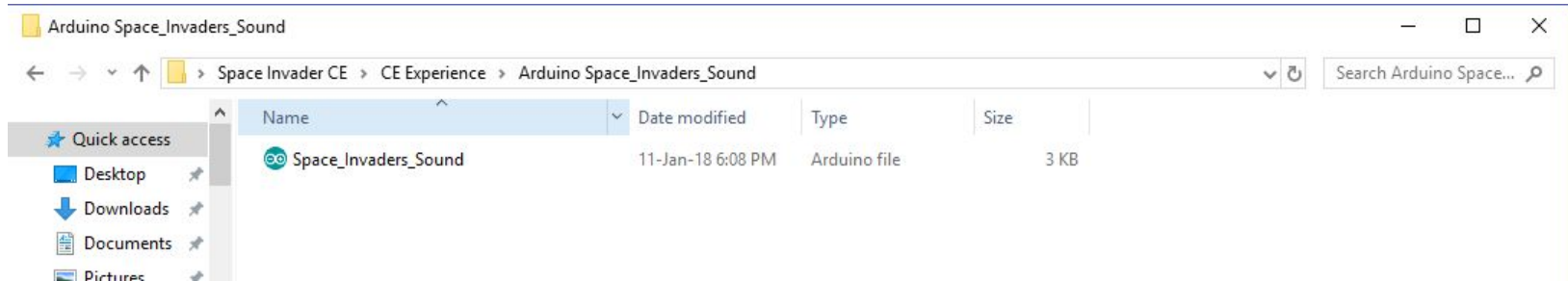
fritzing





# Ejecución del programa

Proceda a conectar su Arduino a la computadora. Luego en la carpeta llamada “Space\_Invaders\_Sound” va encontrar un archivo con extension .ino llamado “Space\_Invaders\_Sound” abra el archivo con arduino.

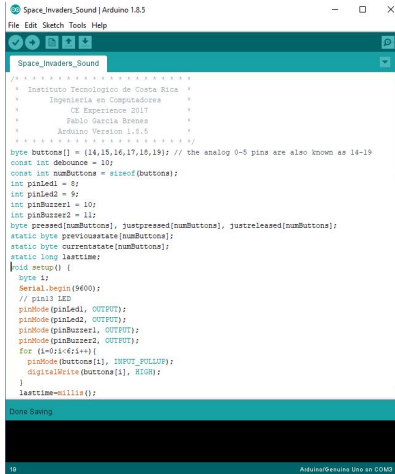


# Ejecución del programa

Se va a abrir el programa con el código necesario para el juego. En la parte superior izquierda se encuentra un botón con un check (utilizado para compilar) y al a par de este botón, se encuentra otro con una flecha que apunta a la derecha (utilizado para cargar el código en el arduino).

En la parte inferior de negro se encuentra la consola (Si hay algún error se mostrará ahí). Encima de la consola tenemos una barra que nos indica la tarea que está llevando a cabo el arduino.

Compile el programa. Si presenta algún error llame a un asistente. A continuación se da un ejemplo de mensaje de error.



```
Space_Invaders_Sound | Arduino 1.8.5
File Edit Sketch Tools Help

Space_Invaders_Sound

/*
 * Instituto Tecnológico de Costa Rica
 * Ingeniería en Computadores
 * UE Experiencia 2017
 * Pablo García Brenes
 * Arduino Version 1.8.5
 */

// =====

byte buttons[] = {14,15,16,17,18,19}; // the analog 0-5 pins are also known as 14-19
const int debounce = 10;
const int numButtons = sizeof(buttons);
int pinLed1 = 8;
int pinLed2 = 9;
int pinBuzzer1 = 10;
int pinBuzzer2 = 11;
byte pressed[numButtons], justPressed[numButtons], justReleased[numButtons];
static byte previousState[numButtons];
static byte currentState[numButtons];
static long lastTime;

void setup() {
  byte i;
  Serial.begin(9600);
  // pin1 LED
  pinMode(pinLed1, OUTPUT);
  pinMode(pinLed2, OUTPUT);
  pinMode(pinBuzzer1, OUTPUT);
  pinMode(pinBuzzer2, OUTPUT);
  for (i=0;i<numButtons;i++)
    pinMode(buttons[i], INPUT_PULLUP);
  digitalWrite(buttons[0], HIGH);
  lastTime=millis();
}

void loop() {
  byte i;
  Serial.println("Loop");
  // pin1 LED
  digitalWrite(pinLed1, LOW);
  digitalWrite(pinLed2, LOW);
  digitalWrite(pinBuzzer1, LOW);
  digitalWrite(pinBuzzer2, LOW);
  for (i=0;i<numButtons;i++)
    digitalWrite(buttons[i], HIGH);
  lastTime=millis();
}
```

Compilar

Ejecutar

```
private static  
int happy  
MyDesktop  
desktop  
happine  
return  
}
```

Error

```
Space Invaders_Sound | Arduino 1.8.5  
File Edit Sketch Tools Help  
Space Invaders_Sound $  
/*  
 * Instituto Tecnológico de Costa Rica  
 * Ingeniería en Computadores  
 * CE Experience 2017  
 * Pablo García Brenes  
 * Arduino Version 1.8.5  
 */  
  
byte buttons[] = {14,15,16,17,18,19}; // the analog 0-5 pins are also  
const int debounce = 10;  
const int numButtons = sizeof(buttons);  
int pinLed1 = 8;  
int pinLed2 = 9;  
int pinBuzzer1 = 10;  
int pinBuzzer2 = 11;  
byte pressed[numButtons], justpressed[numButtons], justreleased[numButtons];  
static byte previousstate[numButtons];  
static byte currentstate[numButtons];  
static long lasttime;  
void setup() {  
  byte i;  
  Serial.begin(9600);  
}
```

expected ';' or ',' before 'int'  
expected ',' or ';' before 'int'

Copy error messages

12 Arduino/Genuino Uno on COM3

wallpaper) {

Consola

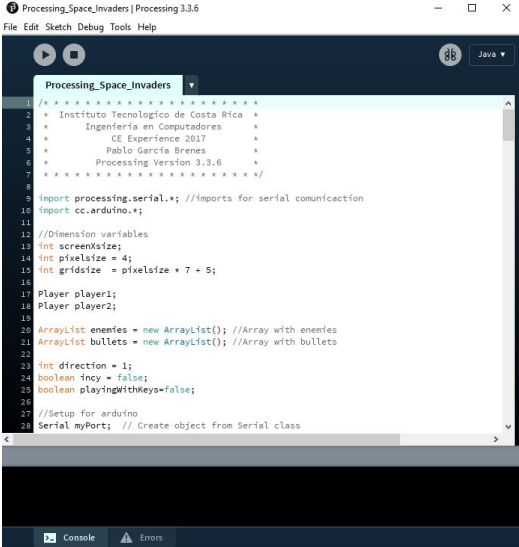


# Ejecución del programa

Si no hay ningún error. En vez del mensaje mostrado en la imagen anterior dirá “Done compiling.”. Ahora proceda a ejecutar el programa.

Luego abra la carpeta llamada “Processing\_Space\_Invaders”. Adentro hay un archivo llamado “Processing\_Space\_Invaders” ejecute el programa utilizando processing.

Se va a abrir el programa junto con el código para el juego. Similar a Arduino en la esquina superior izquierda hay un botón de play (el cual ejecuta el programa) y un botón de stop. En la parte inferior hay una pantalla negra para la consola.



```
Processing_Space_Invaders | Processing 3.3.6
File Edit Sketch Debug Tools Help

Processing_Space_Invaders
1 /* *****
2  * Instituto Tecnológico de Costa Rica *
3  * Ingeniería en Computadores *
4  * CE Experience 2017 *
5  * Pablo García Brenes *
6  * Processing Version 3.3.6 *
7  * ***** */
8
9 import processing.serial.*; //imports for serial communication
10 import cc.arduino.*;
11
12 //Dimension variables
13 int screenSize;
14 int pixelSize = 4;
15 int gridSize = pixelSize * 7 * 5;
16
17 Player player1;
18 Player player2;
19
20 ArrayList enemigos = new ArrayList(); //Array with enemies
21 ArrayList bullets = new ArrayList(); //Array with bullets
22
23 int direction = 1;
24 boolean incy = false;
25 boolean playingWithKeys=false;
26
27 //Setup for arduino
28 Serial myPort; // Create object from Serial class
```



Processing\_Space\_Inaders | Processing 3.3.6

File Edit Sketch Debug Tools Help

Processing\_Space\_Inaders

```
1  / * * * * *
2  * Instituto Tecnológico de Costa Rica *
3  * Ingenieria en Computadores *
4  * CE Experience 2017 *
5  * Pablo García Brenes *
6  * Processing Version 3.3.6 *
7  * * * * */
8
9  import processing.serial.*; //imports for serial comunicaction
10 import cc.arduino.*;
11
12 //Dimension variables
13 int screenXsize;
14 int pixelsize = 4;
15 int gridsize = pixelsize * 7 + 5;
16
17 Player player1;
18 Player player2;
19
20 ArrayList enemies = new ArrayList(); //Array with enemies
21 ArrayList bullets = new ArrayList(); //Array with bullets
22
23 int direction = 1;
24 boolean incy = false;
25 boolean playingWithKeys=false;
26
27 //Setup for arduino
28 Serial myPort; // Create object from Serial class
```

Console Errors



# Juego



Una vez que se presione el botón play, el juego comienza inmediatamente por lo que ambos jugadores deben estar preparados. Proceda a presionar el boton de play.

Se va abrir una ventana como la que se muestra en la imagen. Si tiene algún problema, llame a un asistente.

