

MATERIA DE SISTEMAS EMBEBIDOS

LABORATORIO 3 - JUEGO DE ATARI

Carlos Fernando Guerrón Pantoja

18 de enero de 2021

1. Introducción

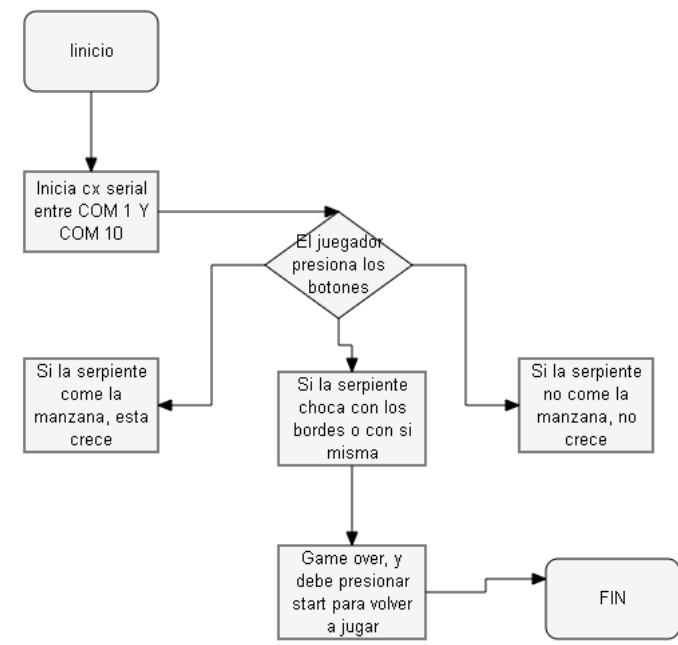
El presente laboratorio es sobre la creación de un juego en la plataforma de Processing y posteriormente este conectarlo al arduino para que se pueda controlar el juego mediante un hardware, este hardware esta compuesto por el arduino y por una carcasa que se diseño para este en el programa de Fusión 360, y aquí se pone los botones para controlar al juego.

Para este caso se diseño el juego de la culebrita, que es mundialmente conocido y es uno de sus mas referentes de Atari, aquí se emplea varias funciones en el diseño del programa, y se aplica lo visto en clase y otras consultas que se hicieron para poder realizar dicho programa.

2. Diseño del Sistema

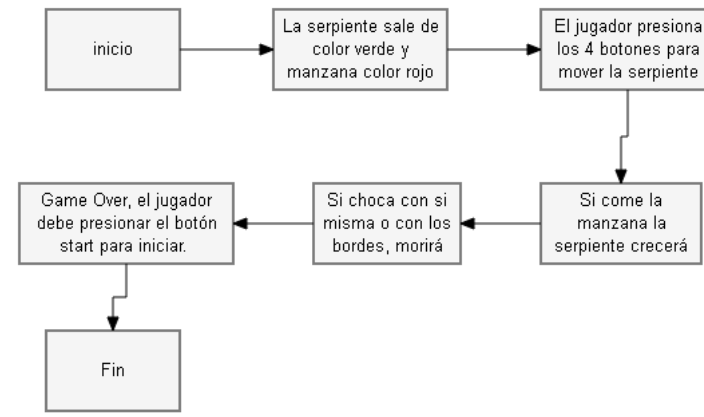
2.1. Diagrama de Flujo

Figura 1: Diagrama de flujo



2.2. Diagrama de bloques

Figura 2: Diagrama de bloques



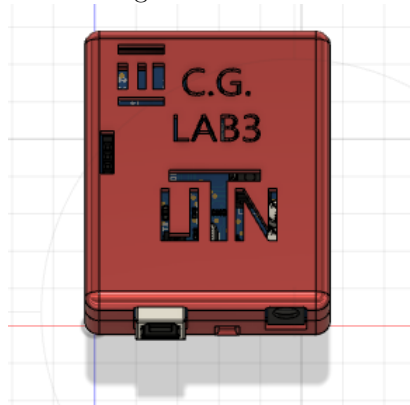
3. Desarrollo

3.1. Simulación

A continuación se muestran los resultados obtenidos al momento de realizar la carcasa para el arduino en fusión 360.

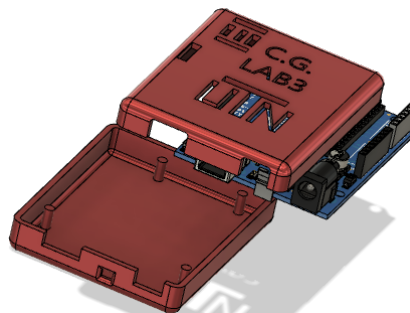
La figura 3 nos muestra la carcasa completa que se realizó en la aplicación de fusión 360.

Figura 3: Carcasa



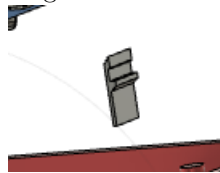
En la figura 4 se muestra la estructura interna de la carcasa, se puede observar que se dejaron 4 patas para meter el arduino en esos orificios y así quede fijo en nuestra caja.

Figura 4: Estructura de la carcasa



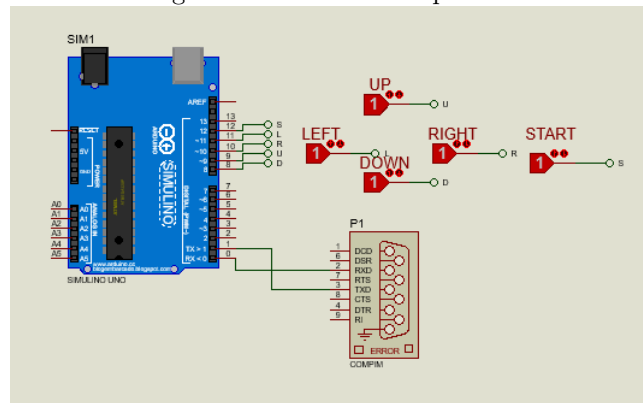
Se diseñó una vincha para que sujete las dos partes de la caja y así se pueda abrir y cerrar.

Figura 5: Vincha



Se presenta la simulacion que se realizo en proteus con los 4 botones que se uso para hacer el juego

Figura 6: Simulacion en proteus



4. Análisis de Resultados

Los archivos que se emplearon para realizar este laboratorio estan ubicados en el repositorio de GitHub: https://github.com/cfguerronp/SistemasEmbebidos_Laboratorios/tree/main/GuerronCarlos_Laboratorio3 Se muestra el funcionamiento, como se puede observar se tiene en el proteus 5 botones, 4 son para controlar a la culebrita (verde), y el otro botón es de start es para iniciar a jugar nuevamente cuando la serpiente choque.

Figura 7: Juego funcionando

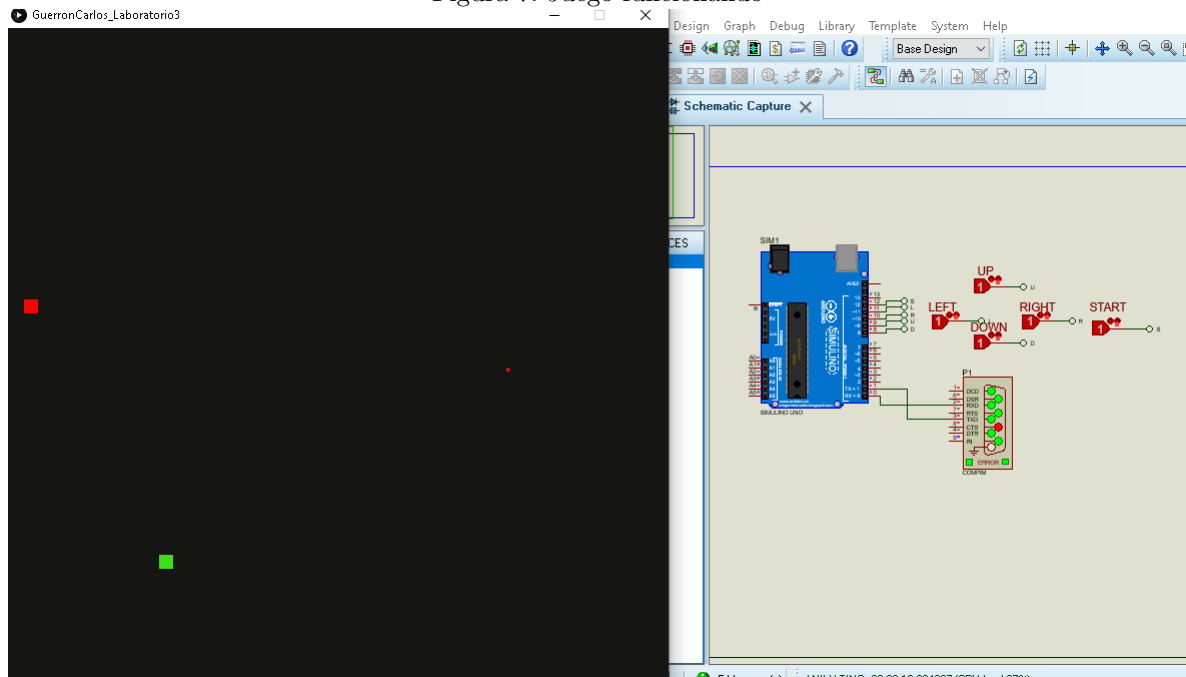


Figura 8: Código processing

```

GuerronCarlos_Laboratorio3
8  */
9  import processing.serial.*;
10 int filas = 44; // numero de filas
11 int columnas = 44; // numero de columnas
12 int bs=15; //tamaño de los cuadritos que la serpiente
13 String dato;
14 ArrayList<Integer> posX = new ArrayList<Integer>();
15 ArrayList<Integer> posY = new ArrayList<Integer>();
16
17 int direccion = 1; //para la direccion
18 int[] dx={0, 0, -1, 1}; //para la direccion de la serpiente
19 int[] dy={-1, 1, 0, 0}; //para ir de arriba y abajo
20
21 int appleX; //para la manzana pos x
22 int appleY; //pos y
23 boolean gameOver = false; // para el fin de juego
24 Serial port;
25 void setup(){
26   port= new Serial(this, "COM10",9600);
27   size(660,660); //tamaño de la ventana
28   posX.add(10); //empieza las pos x
29   posY.add(10); //pos y
30   frameRate(5); //para la velocidad de la serpiente
31   appleX= (int)random(0, 44); //para poner en un lugar
32   appleY= (int)random(0, 44); //pos y

```

Figura 9: Código processing

```

GuerronCarlos_Laboratorio3
25 void draw(){
26   background(#151513); //color de fondo
27   if(gameOver == true){
28     fill(155); //mensaje para que presione tecla
29     textSize(20);
30     text("GAME OVER",300, height/3);
31     text("PRESS START TO PLAY",250, height/2);
32     appleX =-1; //para desaparecer la manzana
33     appleY =-1;
34     posX.clear();
35     posY.clear();
36     posX.add(-10);
37     posY.add(-10);
38   }else{
39     stroke(#151513); //color de linea
40     for(int i=0; i<filas; i++){
41       line(0, i*bs, width, i*bs); //para dibujar
42     }
43     for(int j=0; j<columnas; j++){
44       line(j*bs, 0 ,j*bs, height); // para dibujar
45     }
46   }
47   key();
48   move();
49   eat();

```

Figura 10: Código Arduino

```

GuerronCarlos_Laboratorio3
9 void setup() {
10   Serial.begin(9600);
11   pinMode(8,INPUT);
12   pinMode(9,INPUT);
13   pinMode(10,INPUT);
14   pinMode(11,INPUT);
15   pinMode(12,INPUT);
16 }
17
18 void loop() {
19   if(digitalRead(11)==LOW){
20     delay(250);
21     Serial.print("A");
22   }
23   if(digitalRead(10)==LOW){
24     delay(250);
25     Serial.print("B");
26   }
27   if(digitalRead(9)==LOW){
28     delay(250);
29     Serial.print("C");
30   }
31   if(digitalRead(8)==LOW){
32     delay(250);
33     Serial.print("D");
34   }
35   if(digitalRead(12)==LOW){
36     delay(250);
37     Serial.print("E");

```

5. Conclusiones y Recomendaciones

■ Conclusiones

Fusión 360 es una herramienta muy útil para el estudiante, ya que se puede realizar distintos diseños en 3D, en este caso se realizó una carcasa para el arduino uno, si realizamos un circuito también le podemos realizar cualquier tipo de carcasa para que nuestros proyectos sean mas prácticos.

Fusion 360 nos permite exportar nuestros diseños para que se pueda imprimir en una impresora 3D lo que es muy funcional y ayuda mucho al estudiante.

Processing nos ayuda a realizar las interfaces de nuestros programas de arduino, estos dos van de la mano y así en este laboratorio se hizo un juego y puede ser controlado con el arduino y la aplicación funcionando de buena manera.

Las distintas funciones que se emplearon en la realización del juego se aprendieron en clase y también con aprendizaje autónomo ya que Processing es una herramienta muy amplia y tiene muchas funcionalidades para que pueda funcio-

nar de mejor manera.

Para que se pueda conectar nuestro arduino de proteus con el programa de Processing es necesario crear una conexión de los puertos COM, esto se hace mediante la aplicación VSPE, con esto ya podemos conectar estos dos puertos y hacer que funcione nuestro juego.

■ Recomendaciones

Al momento de realizar nuestras figuras en fusión 360 se debe trabajar con paciencia hasta poder entender el funcionamiento de este programa, ya que al inicio es muy confuso usar este programa. La programación en processing es recomendable que se haga mediante métodos para tener un programa mas ordenado y si se tiene algún tipo de error se pueda encontrarlo mas fácil.

Las distintas funciones que se tiene en processing deben ser estudiadas y analizadas para entender el funcionamiento de estas y así poder realizar cualquier tipo de interfaz, esto nos sirve para mejorar nuestra manera de programar.