

**Introducción a la Programación de  
Microcontroladores con Tecnologías  
Libres:**

**Mini Arcade:** Tetris

Integrantes: Sergio Ariel Guzmán, Alan Martínez  
Profesor: José Luis Di Biase  
1er cuatrimestre 2019



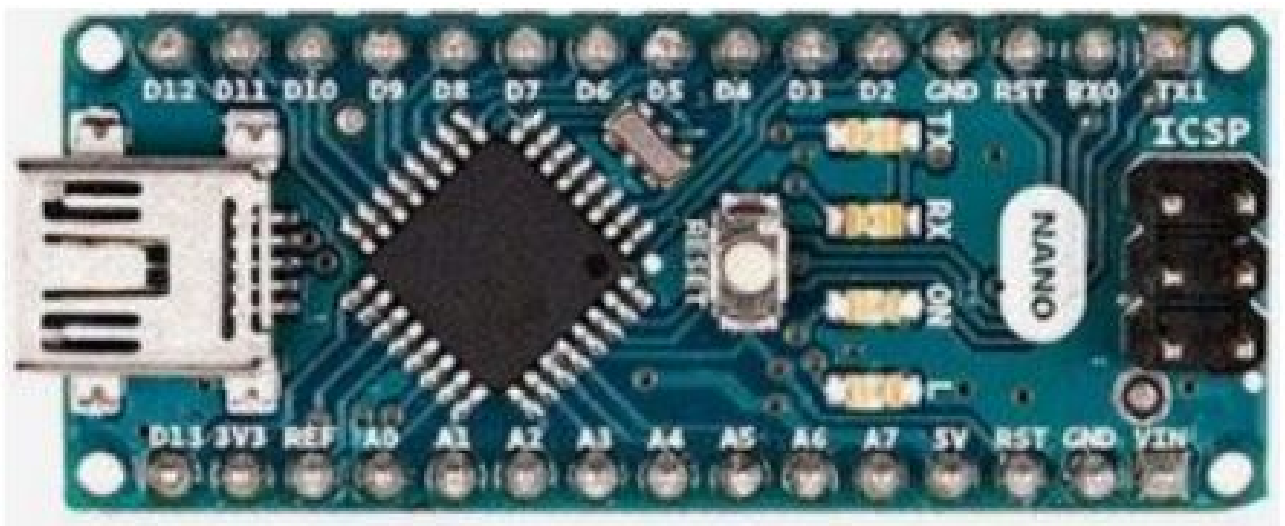
## Descripción:

Este proyecto consiste en replicar un juego arcade clásico de los 80, el tetris. El proyecto fue montado en una carcasa de madera fabricada manualmente de un aproximado de 13cm x 6cm y adaptado con una batería de 9v para hacer del juego una versión portátil la cual denominamos Mini Arcade.

El mini arcade consta de un modulo jostick analogico que le proporciona mayor comodidad para realizar los movimientos del juego, dos botones: uno con la funcionalidad de iniciar el juego y otro con la de acelerar la velocidad con las que caen las piezas, un display por donde se visualiza el progreso del juego y un interruptor para encender o apagar el juego.

## Materiales:

### Arduino nano V3.0 Atmega 328



## Display Lcd Arduino Pic 1.44 Serial 128x128 Spi Color Tft



## Modulo Joystick Analogico



## Botón pulsador



## Bateria 9v



## Conector para batería



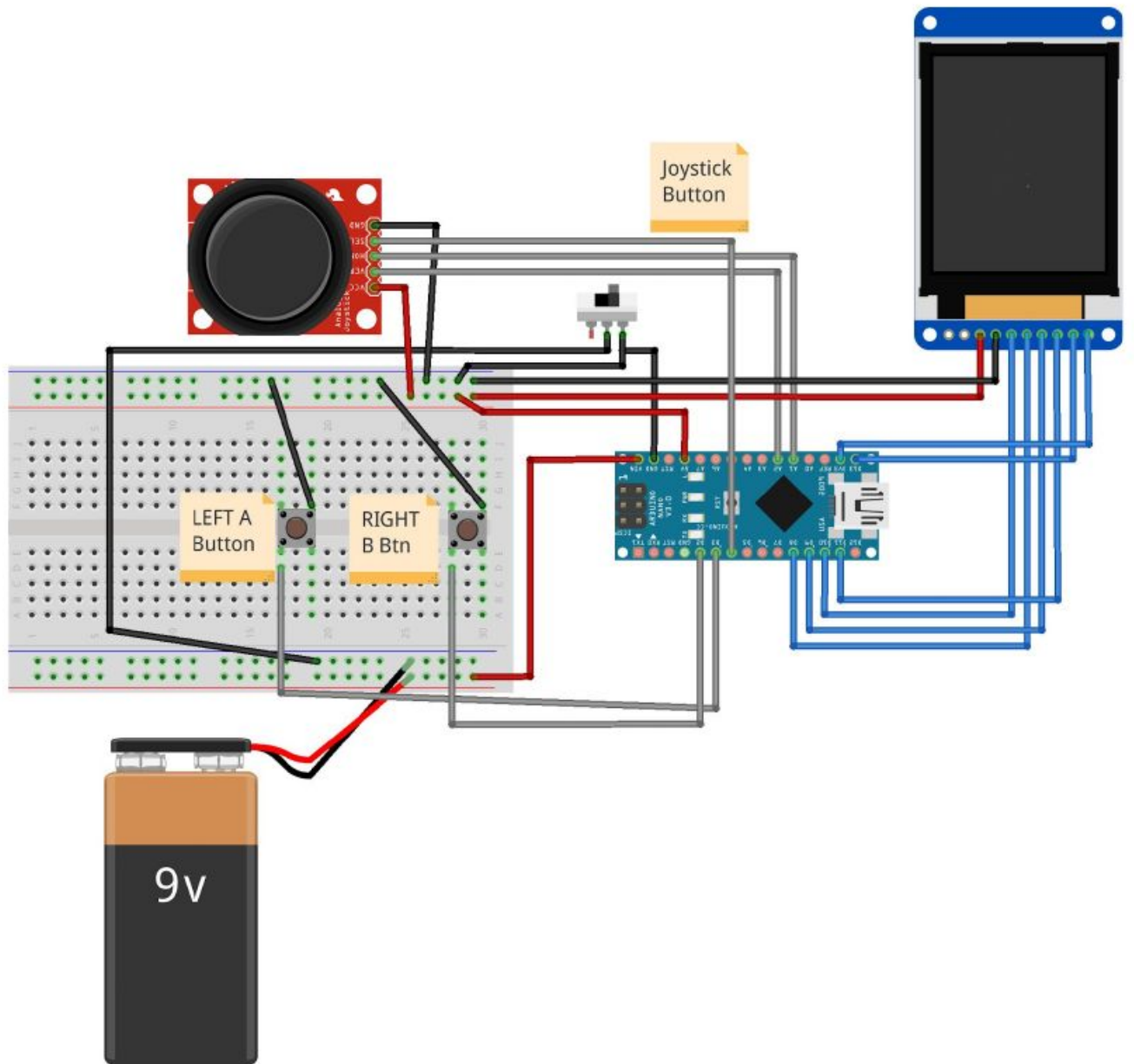
## Interruptor



## Cables



# Esquemático:



## Montaje:

El primer paso es armar el circuito, en la siguientes tablas hay un mapeo de las conexiones de los componentes principales a los pines del arduino.

### Modulo Jostick analogico

Arduino Pin	Joystick Pin
Vcc 5V	Vcc 5V
Gnd	Gnd
A1	VRx
A0	VRy
D4	SW

### Display TFT Color

Arduino Pin	TFT Pin
5V/Vcc	Vcc
Gnd	Gnd
8	RESET
9	AO - D/C
10	CS
11	SDA (MOSI)
12 Niet Nodig	SDO (MISO) Niet nodig
13	SCK
3.3V	LED

Una vez montado todo el circuito en una protoboard el próximo paso es cargar el código al arduino nano. Para esto necesitamos conectar el arduino nano a una computadora a través de un cable mini usb, pero antes de realizar la carga propiamente dicha del juego a la placa necesitamos agregar una librería que se encarga de proveer los controladores del display lcd tft desde el ide de arduino.

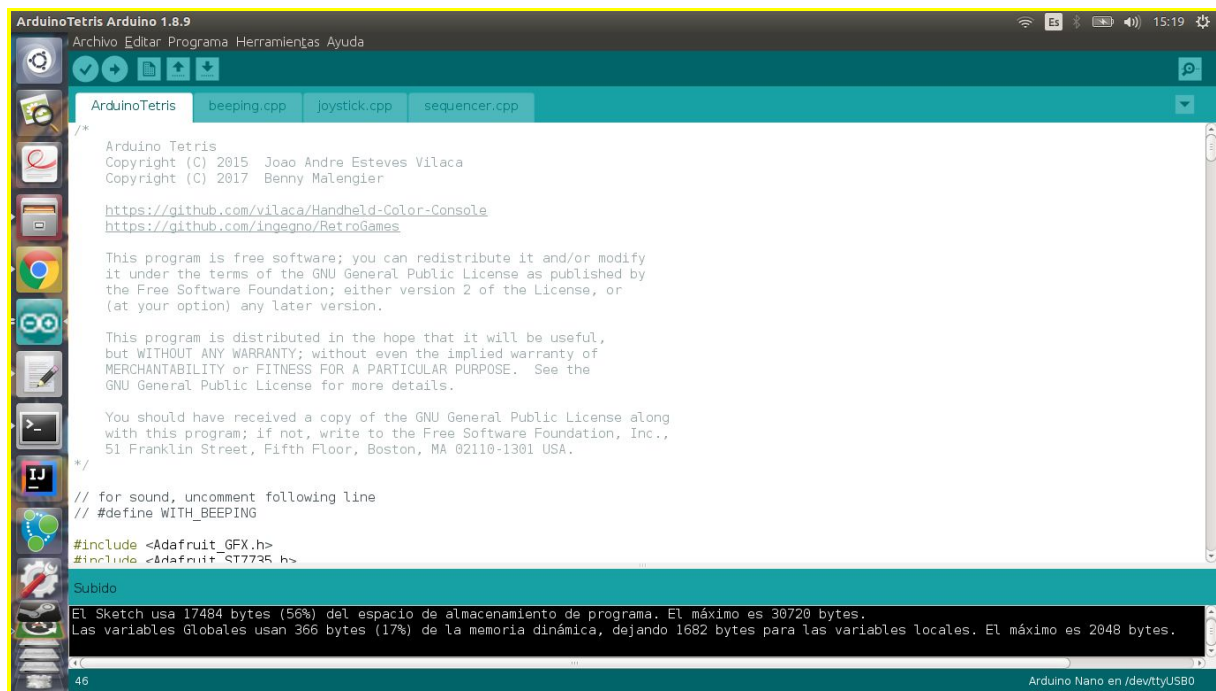
Adafruit\_GFX

<https://github.com/adafruit/Adafruit-GFX-Library>

Adafruit\_ST7735

<https://github.com/adafruit/Adafruit-ST7735-Library>

Una vez descargada la agregamos al proyecto a través del ide de arduino.



```
ArduinoTetris Arduino 1.8.9
Archivo Editar Programa Herramientas Ayuda

ArduinoTetris beeping.cpp joystick.cpp sequencer.cpp

/*
  Arduino Tetris
  Copyright (C) 2015 Joao Andre Esteves Vilaca
  Copyright (C) 2017 Benny Malengier

  https://github.com/vilaca/Handheld-Color-Console
  https://github.com/ingegno/RetroGames

  This program is free software; you can redistribute it and/or modify
  it under the terms of the GNU General Public License as published by
  the Free Software Foundation; either version 2 of the License, or
  (at your option) any later version.

  This program is distributed in the hope that it will be useful,
  but WITHOUT ANY WARRANTY; without even the implied warranty of
  MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
  GNU General Public License for more details.

  You should have received a copy of the GNU General Public License along
  with this program; if not, write to the Free Software Foundation, Inc.,
  51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.
*/

// for sound, uncomment following line
// #define WITH_BEEPING

#include <Adafruit_GFX.h>
#include <Adafruit_ST7735.h>

Subido
El Sketch usa 17484 bytes (56%) del espacio de almacenamiento de programa. El máximo es 30720 bytes.
Las variables Globales usan 366 bytes (17%) de la memoria dinámica, dejando 1682 bytes para las variables locales. El máximo es 2048 bytes.

46 Arduino Nano en /dev/ttyUSB0
```

Verificamos el código y presionamos el botón cargar.

De salir todo bien el juego ya estaría corriendo en nuestro arduino.



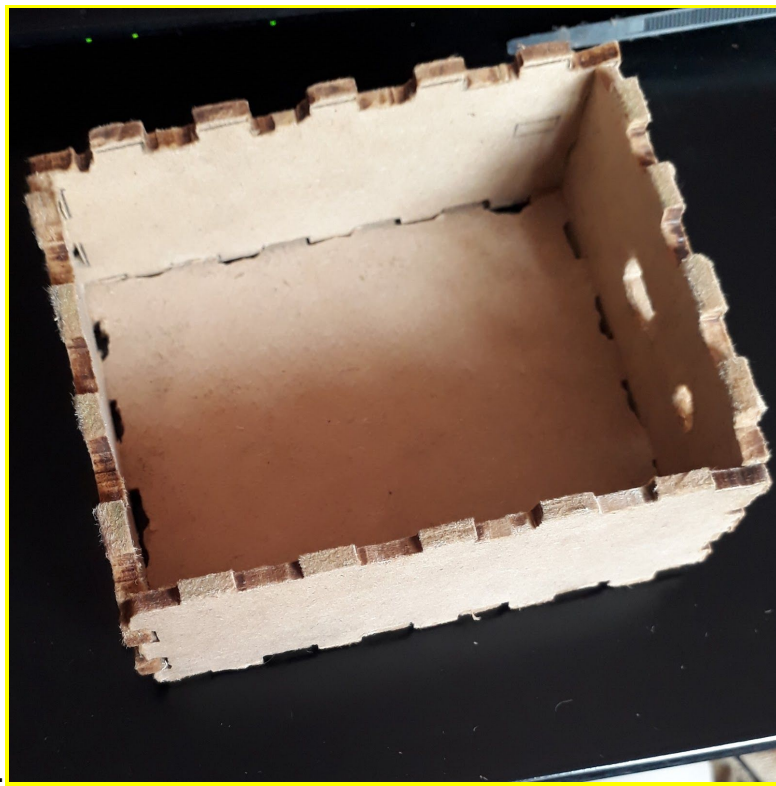
El último paso es el de montar la carcasa y soldar los componentes del circuito.

### **Carcasa**

Para armar la carcasa decidimos utilizar fibrofacil que es más barato que otros materiales y más fácil de manipular.

En el proyecto incluimos un archivo .svg con el molde de la carcasa. Este archivo se puede utilizar para realizar cortes láser y tener una mayor prolijidad. Pero de no disponer de una cortadora láser se puede realizar manualmente con una pequeña sierrita y un cutter. Para realizar el corte manualmente basta con imprimir el archivo .svg como imagen y calcar el dibujo del molde prolijamente en una plancha de fibrofacil con papel carbón.

Una vez dibujado el molde en la plancha de fibrofacil se procede cuidadosamente a cortarla siguiendo cada detalle de las líneas



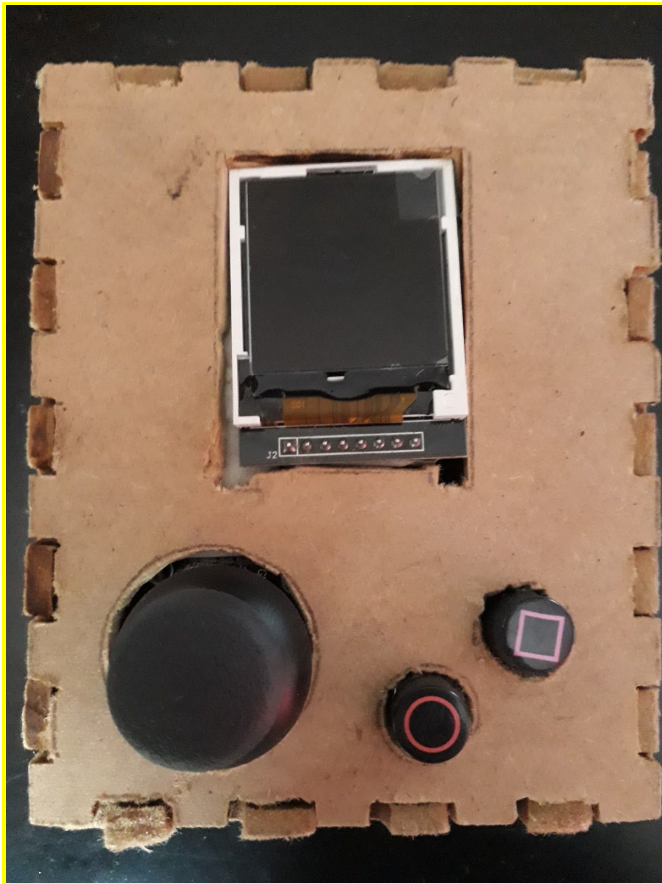
dibujadas.

Esto puede ser un trabajo arduo y tedioso ya que los bordes de las piezas de la carcasa no son rectos sino más bien dentados, esto nos facilitara el ensamble final, una vez cortadas todas las piezas

armarlo seria cuestion de ensamblarlo respetando los bordes dentados por lo que cada pieza encastra con la siguiente luego aplicamos pegamento en los pliegues para lograr mayor estabilidad.

### **Montaje final:**

El último paso es montar dicho circuito terminado y colocar cada parte en su respectivo lugar de la carcasa. Para esto utilizamos cinta de papel y pegamento.



## Problemas:

### Arduino generico

La primera dificultad a la que nos enfrentamos en el proyecto fue que al comprar la placa arduino nano adquirimos una copia que requería que descargáramos drivers hechos en china, luego ignorando la existencia de estas nos encontramos con un arduino con fallas a la hora de cargar los programas, dicho problema no siempre pasa, pero algunos tienen la falla en el gestor de arranque y por eso no se puede cargar el programa. Si tienes un arduino original se puede reparar el gestor de arranque del mismo pero como no contamos con uno decidimos comprar un arduino nano original.

### Pantalla mas chica:

Otro inconveniente que tuvimos en el proyecto se dio a la hora de comprar el display lcd. Un error a la hora de realizar el pedido online del producto nos trajo como consecuencia una dificultad adicional al proyecto. Adquirimos una pantalla más chica de la que el proyecto necesitaba y habiéndose finalizado los plazos para el cambio del producto la solución que implementamos fue la de modificar el código del juego para adaptarlo al display que teníamos pero no se pudo corregir al 100 por ciento.

### Analogico con fallas:

Otra dificultad que tuvimos también relacionada a la mala calidad de los componentes genéricos se dio con el módulo joystick analógico. Este tenía una pequeña falla, que hacía que las piezas

del juego estén siempre en el margen izquierdo. Como si se estaría presionando en esa dirección.

Una vez detectada esta falla decidimos como solución reemplazar el componente.

Cortar la carcasa:

El material elegido para realizar la carcasa del mini arcade era muy sensible por lo que cortarla demandó un arduo trabajo y aun así no se obtuvo un resultado cien por ciento satisfactorio en cuanto a la prolijidad esperada.

## **Código Fuente**

<https://github.com/fuser1988/MiniArcadeTetris.git>

## **Referencias:**

<https://github.com/ingegno/RetroGames>