Bitrix

El informe deberá tener como mínimo:

1. Caratula con nombre del proyecto y datos de los integrantes
2. Objetivo
3. Breve descripción (esquemas, diagramas, etc.)
4. Principio de funcionamiento y fundamentos teóricos (ecuaciones, graficos,etc)
5. Circuitos electronics (esquemáticos)
6. Modelo de circuito impreso (PCB) lado de cobre y componentes
7. Lista de componentes, costos
8. Posibles aplicaciones
9. Indice

# Introducción

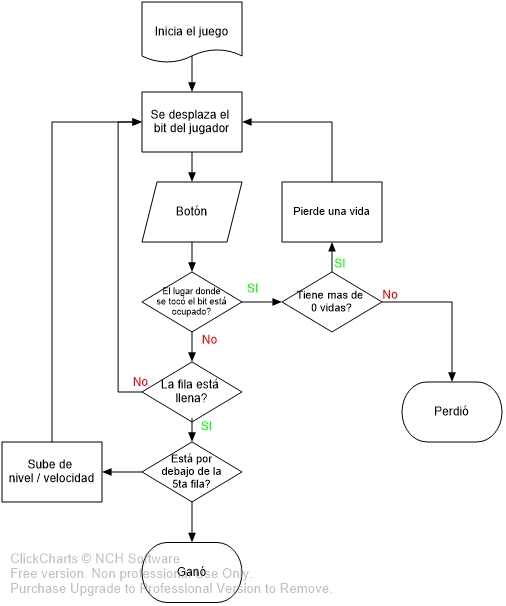
Este proyecto propone crear un Tetris “de un bit”, a diferencia del Tetris común, este no tendrá formas complejas como la L,T, el cubo o el palo; únicamente dispondrá de un único punto (un led) que ira de izquierda a derecha en lo más alto de la pantalla. Una vez el jugador apriete un botón, se dejará caer este bit a lo más profundo de la pantalla, el objetivo es que el jugador vaya alineando horizontalmente bit por bit como se muestra en el siguiente gráfico.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| 1 |  |  | 9 |

Una vez la fila es completada el jugador pasará de nivel y deberá proseguir a llenar la siguiente fila, a medida que el desplazamiento del bit superior va aumentando de velocidad.

El desafío del juego se encuentra en que el jugador **NO** debecomenzar a llenar la siguiente fila sin haber completado la anterior (ejemplo en el grafico debajo). Si esto llegara a suceder el jugador perderá una de sus vidas.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | |  | |  | |  | |
|  |  | |  | |  | |  | |
|  |  | |  | |  | |  | |
|  |  | |  | |  | |  | |
|  |  | |  | |  | | El jugador comenzó a completar la 2da | |
|  |  | |  | |  | | Fila sin completar la 1ra | |
|  | |  | |  | |  | |
| 0 | |  | |  | | 8 | |



Debido a que a medida que suba de nivel se aumentará la velocidad de desplazamiento, al jugador se le hará más difícil completar adecuadamente las filas.

Para tener una mayor comprensión del funcionamiento y de lo que se encargará el futuro programa, se realizó el siguiente diagrama de flujo simplificado

# Principio de funcionamiento

## Matriz y multiplexado

Debido a la limitada cantidad de pines disponibles en el microcontrolador y a la poca energía que estos pueden entregar en su salida, se debió utilizar técnicas de multiplexado, tanto para el control de la matriz 6x4 como para los displays de vida restantes y nivel.

Para esto se utilizó uno de los timer del ATMEGA 328p en modo de comparación con un prescaler de 1024 y un valor de comparador de 3. Con esto en una frecuencia de aproximadamente 16MHz nos da una función de multiplexado que se activa cada:

O sea, con una frecuencia de:

Si bien esto está muy por encima de la frecuencia de 50Hz necesaria para que el ojo humano no distinga el prendido y apagado, se optó por esta velocidad ya que el programa debe realizar el maskeo completo de la matriz y modificar los valores de los displays dentro de estos intervalos.

Por lo que la frecuencia real de multiplexado completo es el intervalo entre cada activación de la función dividido cada led de la matriz:

Si bien sigue siendo más de 4 veces la velocidad requerida, preferimos utilizarla de todas maneras ya que no afecta el funcionamiento del juego y a la vez conseguimos una visualización del display más limpia que a frecuencias más bajas.

## Display de vidas y nivel

Como se mencionó en el punto anterior, los displays son controlados en parte por multiplexación para mostrar las vidas restantes y el nivel actual.

Pero se presentó la problemática de la cantidad de pines limitados del ATMEGA328p por lo que no era posible controlar directamente los 7 segmentos a través de él utilizando una decodificación por software. En cambio, se decidió utilizar un CD4511 de intermediario.

El CD4511 es un decodificador de binario a 7 segmentos, lo que quiere decir que una vez conectado las entradas correspondientes y las salidas a su respectivo segmento, en el display se vera el numero ingresado en formato binario en las entradas.

De esta forma se ahorran 3 pines que pueden ser utilizados para otras funciones, el programa se simplifica ya que solo hay que poner las salidas en binario, y mas importante, se libera al microcontrolador de la carga de tener que alimentar los 7 segmentos del display, ya que estos serán controlados por el mismo CD4511, el cual es capaz de soportar mayores cargas.

Cabe aclarar dos cosas:

* Debido a este último aspecto se debe poner resistencias para proteger los segmentos
* El microcontrolador se sigue encargando de controlar que display prender (multiplexado) y que valor mostrar (Se lo indica al CD4511)

## Botón

### Jugador

Debido a que este es un juego de reflejos y velocidad nos pareció que debíamos evitar utilizar la técnica de polling\* para detectar cuando se presiona el botón. En cambio, se tomo ventaja de una de las interrupciones externas del ATMEGA328p, de esta forma el programa dentro del microcontrolador reaccionará de forma casi instantánea cada vez que el botón sea presionado.

\*Consiste en realizar una lectura lo mas frecuentemente posible del estado un pin para determinar si hubo un cambio de estado

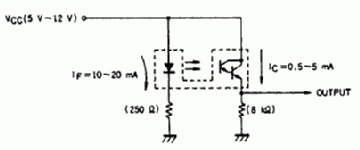
Esta interrupción será activada cuando ocurra un flanco descendente en el pin 4 del microcontrolador.

Para la elección del botón se tomó como principal parámetro la estética y sensación del mismo al presionarlo.

### Monedas

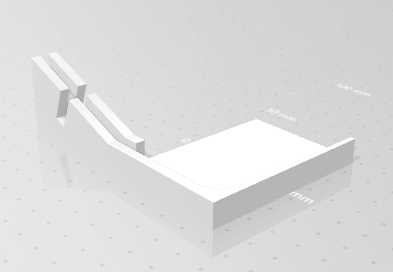
En el caso de la detección de monedas se optó por realizar una ranura en el frontal del gabinete que dirija a una rampa que vaya recolectando las monedas insertadas. Dicha rampa tiene un foto-interruptor incrustado el cual detectará el paso de la moneda.

Se decidió utilizar un PS4005 por su simplicidad de uso.

Como la esquemático interno indica, en su interior no hay más que un LED y un fototransistor

Para hacerlo funcionar de la manera deseada solo se necesita poner unas resistencias de pull up y de limitación de corriente para el LED

El foto-interruptor estará constantemente mandando un 1 en el pin 2 del ATMEGA; cuando pase un objeto entre se el LED y el fototransistor, la señal se irá a masa.

Por medio del software se configuro que esto solo pueda ser activado para iniciar el juego, durante el resto del juego las señales recibidas por el interruptor serán ignoradas. De la misma forma que las señales recibidas del botón del jugador serán ignoradas hasta que el mismo inserte una moneda/ficha.

La rampa fue realizada en AutoCAD para que pueda ser impresa en 3D fácilmente, con la ranura para el foto-interruptor

## Música/amplificación

Para añadirle más carisma al juego se le puso música de fondo. La cual sonará a través de un parlante de 4 Ω

Esto se realizó utilizando la librería Tone de Arduino, la cual fue creada para este mismo propósito. Aun así, el pin del Arduino no es capaz de hacer sonar el parlante utilizado, o por lo menos no con el volumen deseado.

Por esto se decidió hacer una amplificación simple con un optoacoplador (4n28) y un transistor que soporte la carga presentada por el parlante.

Realizando pruebas se encontró que el parlante puede llegar a consumir hasta 350mA aprox. Con una alimentación de 9V. Teniendo en cuenta esto se decidió utilizar un BD137, el cual es un transistor NPN capaz de soportar una corriente de colector de 1,5A.

Como detalle interactivo, la música cambiará su velocidad a medida que el jugador vaya aumentando de nivel.

# Circuitos electrónicos

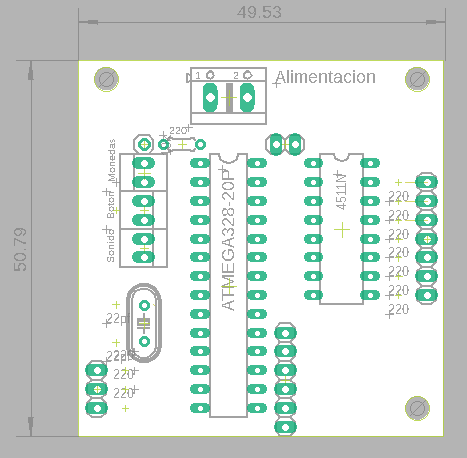
## Placa controladora

## Placa de amplificación de audio

## Placa de displays 7 segmentos

# Diseño PCB

## Placa controladora



## Placa de amplificación de audio

## Placa de displays 7 segmentos

# Lista de componentes