

# **Estructura de Computadores**

Grupo 2B04

Asier Núñez Lozano

Jon Villanueva Errazquin

Beñat Oiartzun Arieta-araunabeña

# **MEMORIA DEL PROYECTO DE DESARROLLO DE UNA APLICACIÓN DE E/S PARA LA NINTENDO DS**

## **1 Introducción**

El proyecto consiste en implementar un juego para una consola Nintendo DS. El objetivo del mismo es llevar a la práctica lo aprendido en el tema de Entrada y Salida, por lo que el juego debe llevar a cabo el control de sus periféricos mediante los dos tipos de sincronización estudiados: por encuesta y por interrupción. Además, debe hacer uso, al menos, del teclado y de la pantalla táctil y debe realizar algún tipo de control de tiempo con temporizadores. Entre las teclas usables, se encuentran todas, excepto la 'x' y la 'y'.

Tenemos a nuestra disposición, tres tipos de temporizadores distintos. Estos presentan, frecuencias distintas entre sí, lo que nos permite configurar la cantidad de interrupciones que podremos hacer en una cantidad de tiempo determinada.

Solo podremos usar la pantalla inferior, para mostrar los grafismos jugables del juego, mientras que la pantalla superior la usaremos para mostrar tiempos o muestras de la consola.

## 2 Descripción del juego

Nuestro juego consiste en una pequeña carrera de coches. La modalidad de la carrera es "Drag Race", lo que significa que los coches solo avanzan, no giran. La carrera consiste en empezar en el inicio y acabar en la meta. Siempre participan 4 coches, estos pueden ser controlados por jugadores reales o a lo que llamamos inteligencia artificial (IA), aunque de inteligencia poco tiene.

El juego comienza con un menú, donde podremos elegir la cantidad de jugadores "reales" queremos que jueguen. Esta opción la podremos elegir tocando en la opción específica de la pantalla táctil. Esto nos dirigirá al siguiente menú, donde podremos elegir la dificultad de la partida que pretendemos jugar. La dificultad repercute en la cantidad de píxeles que avanzará la IA por cada tick del temporizador.

Tras esto, ya tenemos todo listo, nos encontramos ante una calle vista desde un plano cenital donde vemos cuatro carriles en la carretera. Cada coche estará en la parte izquierda de la pantalla, dentro de su carril correspondiente. Una vez cargada la pantalla de la partida, inicia un contador de 2 segundos (en este caso el contador no es visible) y cuando este llegue a cero, empieza la carrera. Cada jugador deberá presionar el botón que le corresponde para que el coche avance. Cuanto más rápido se presione el botón, más rápido avanzará el coche.

El jugador 1 deberá presionar el botón A

El jugador 2 deberá presionar el botón B

El jugador 3 deberá presionar el botón ARRIBA

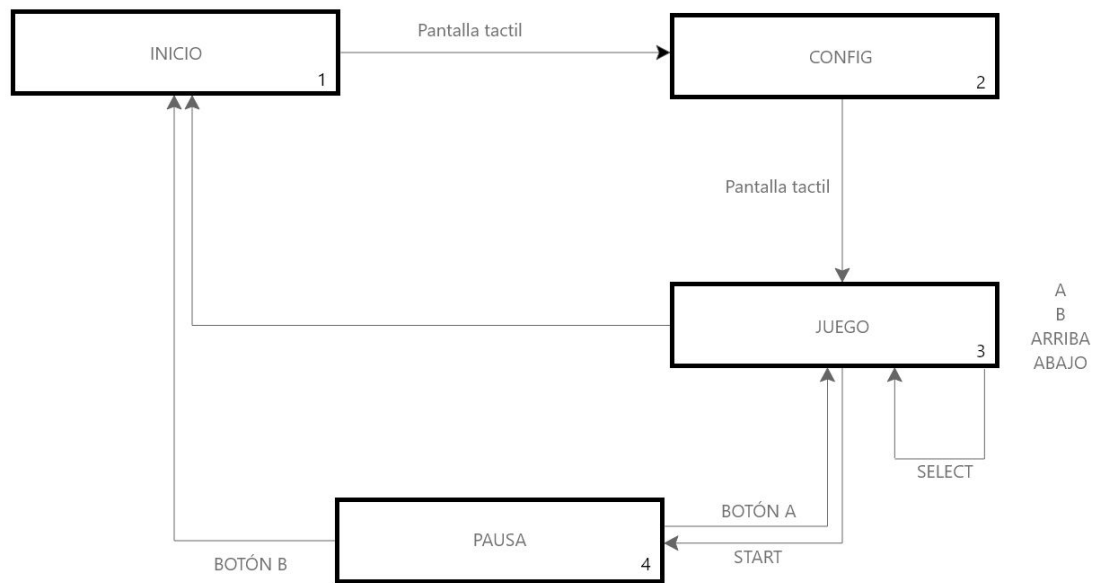
El jugador 4 deberá presionar el botón ABAJO

Como ya hemos mencionado, la carrera finaliza cuando todos lleguen al final de la calle, aunque ganará el primero que llegue.

En cualquier momento se podrá pausar el juego, para ello tendremos dos menús de pausa. Uno nos permitirá parar el juego y reanudarlo cuando queramos y el otro nos manda directamente al menú principal. A estos menús se puede acceder mediante los botones <START> y <SELECT>, que están controlados por interrupciones.

Después de terminar la partida, nos redirigirá a la pantalla de menú de inicio.

### 3 Autómata



1. **INICIO**. En esta pantalla se seleccionará la cantidad de jugadores (1-4) mediante el uso de la pantalla táctil mediante encuesta.
2. **CONFIG**. Tras seleccionar la cantidad de jugadores en la pantalla anterior, mediante el uso de la pantalla táctil se seleccionará la dificultad de los jugadores controlados por la máquina.
3. **JUEGO**. Aquí comenzará el juego, que consiste en presionar repetidamente un botón para avanzar (A, B, ARRIBA, ABAJO los cuales funcionarán por encuesta) y competir con el resto de jugadores. Estando en esta pantalla, si presionamos el botón **START**, el cual funcionará mediante interrupciones, se desplegará un menú de pausa y el pulsar **SELECT**, el cual también funciona mediante interrupciones se pausará el juego. Al finalizar el juego se volverá automáticamente a la pantalla de **INICIO**. El temporizador por medio de interrupciones servirá para controlar a los jugadores que falten para llegar a 4, es decir, si vamos a jugar 2 personas, los otros 2 jugadores estarán controlados con el temporizador. El temporizador, servirá como referencia para la dificultad. Elegiremos una cantidad de tik-s determinada (tempo, que puede ser un segundo o lo que creamos conveniente), que nos parezca equilibrado para jugar. Por cada "tempo", los jugadores controlados por el temporizador, avanzarán x píxeles. Aquí es donde entra en juego la dificultad, según cuál se elija, la cantidad de píxeles que avanzarán será mayor o menor.

4. PAUSA. En el menú de pausa se podrá volver al juego y continuarlo como estaba antes de darle al pausa con la tecla A. Y mediante la tecla B se volverá al INICIO.
5. También se podrá pulsar en cualquier momento la tecla START (interrupción) para pausar el juego y volver a donde nos encontrábamos con la tecla A.

## 4 Desarrollo del proyecto

### Sprites.c

- MostrarCar (int indice, int x, int y)

Esta función muestra el sprite del coche en las coordenadas definidas por los parámetros x e y. También le da un índice para saber qué sprite es.

- BorrarCar(int indice, int x, int y)

Esta función borra el sprite del coche en las coordenadas definidas por los parámetros x e y.

### Fondos.c

- MostrarJuego()

Esta función muestra el fondo Juego.png.

- MostrarDificultad()

Esta función muestra el fondo Dificultad.png.

- MostrarJugadores()

Esta función muestra el fondo Jugadores.png.

- MostrarPausa()

Esta función muestra el fondo Pausa.png.

## Variables Globales

- extern int estado;

Esta es una variable para saber en qué estado está el juego.

- extern int tik;

Esta es una variable para determinar el número de interrupciones que se ejecutan por parte del temporizador.

- extern int jugadores;

Esta es una variable para saber cuantos jugadores van a participar en el juego.

- extern int timereset;

Esta es una variable para resetear el timer y con este, los sprites que se estén utilizando en el juego.

- extern int dificultad;

Esta es una variable para saber en qué dificultad los jugadores van a jugar el juego.

- extern int reset;

Esta variable se utiliza para que, en caso de encontrarse en un bucle, el juego se vea forzado a salir del bucle.

- extern int tecla;

Esta es una variable para saber que tecla se ha pulsado;

- touchPosition PANT\_DAT;

Esta es una variable para saber en qué posición el jugador toca la pantalla.

## **Main.c**

- delay(int x)

Esta función retrasa la ejecución del código a modo de delay.

- Inicio ()

Esta función se ejecuta en el estado INICIO del juego. Aquí, por medio de la pantalla táctil, se seleccionará la cantidad de jugadores que van a jugar y lo guardará en la variable global jugadores. Al terminar, el estado se cambia a CONFIG.

- Config()

Esta función se ejecuta en el estado CONFIG del juego. Aquí, por medio de la pantalla táctil, se seleccionará la dificultad de los jugadores controlados por el temporizador y lo guardará en la variable global dificultad. Al terminar, el estado se cambia a JUEGO.

- juegoXPlayers() Siendo X el número de jugadores.

Estas son las funciones que, mediante encuesta del teclado, controla los coches. Mediante las variables PUSH, se previene que al mantener pulsada una tecla un coche avance más de una posición.

## **Teclado.c**

- IntTec()

Esta función se ejecutará cada vez que el temporizador marque una interrupción. Lo que, a su vez, se ve influido por el valor del LATCH.

## **Temporizadores.c**

- IntTemp()

Esta función se ejecutará cada vez que el teclado marque una interrupción. Lo que, a su vez, se ve influido por el TECLAS\_CNT.

## **Ruteserv.c**

- HabilitarInterrupciones()

Esta función sirve para habilitar las interrupciones por parte del teclado y del timer. Primero se inhiben todas las interrupciones utilizando la variable IME, después se le adjuntan a la variable IE su nuevo código y por último se vuelven a habilitar las interrupciones.

- Programar Registros De Control()

En esta función se configuran los registros de control de teclado (TECLAS\_CNT) y el temporizador (TIMER0\_CNT y TIMER0\_DAT).

- Definir Vector De Interrupciones()

En esta función se estipula que función se ejecutará cuando se reciba una interrupción, bien sea de teclado o bien de temporizador.

- Inhibir Interrupciones()

Mediante esta función se inhiben las interrupciones del temporizador, Primero se inhiben todas las interrupciones utilizando la variable IME, después se le adjuntan a la variable IE su nuevo código y por último se vuelven a habilitar las interrupciones mediante el IME.



## **5 Conclusiones**

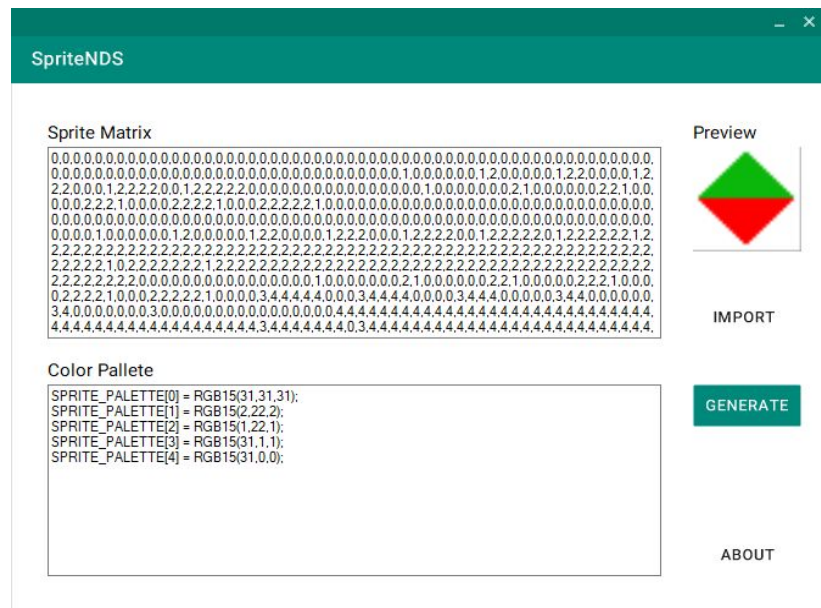
En general, todo ha ido más o menos como esperábamos, sin tener muchas complicaciones, la descripción del juego creemos que es casi idéntico a lo que hemos conseguido implementar.

En un futuro podríamos llegar a implementar más mapas, más modelos de coches e incluso más complejidad jugabilística.

Creemos que la cantidad de horas invertidas en el proyecto escapa a nuestras cuentas, y que no podríamos contabilizarlo. Pero, en general, han sido muchas horas de trabajo.

## 6 Software de generación de Sprites

Además del trabajo requerido, hemos desarrollado una herramienta para la generación de sprites, que a nuestro parecer, facilita el proceso de creación de los mismos. Este software es de código abierto y está disponible en [GitHub](#) para aquellos que quieran utilizarlo o modificarlo.



### Descripción

En esta nueva herramienta de generación de sprites, además de la matriz de colores que ofrecía la anterior versión, se obtiene la paleta de colores, ya sean de 16x16 o de 32x32, listo para copiar y pegar.

### Funcionamiento

Como se puede apreciar en la imagen superior, la interfaz de usuario es muy sencilla. Simplemente hay que accionar el botón "Import", para seleccionar el fichero que se quiere convertir. Tras esto, simplemente pulsar el botón "Generate".

### [Formatos permitidos](#)

### Limitaciones

Recomendamos no usar excesivos colores. Según las pruebas realizadas, la consola no permite sprites con alrededor de 500 colores. Aun así, por prevención, recomendamos de herramientas de edición de imagen para simplificar la paleta de colores.