

PROYECTO NDS

G26

Ander Alonso, Iñigo Bereciartua, Daniel Esteban

Índice

Índice	1
Introducción	2
Descripción	3
Autómata	4
Desarrollo	5
Teclado	5
Temporizador	6
Flags	6
Billetes	7
Recursos utilizados	8
Conclusión	8

Introducción

El proyecto consiste en implementar un juego muy sencillo para una consola Nintendo DS. El objetivo del mismo es manejar los conceptos estudiados en el Tema 3 de Entrada/Salida de la asignatura Estructura de Computadores.

El juego debe: Llevar a cabo el control de sus periféricos mediante distintos tipos de sincronización: por encuesta y por interrupción. Hacer uso, al menos, del teclado y de la pantalla táctil. Realizar algún tipo de control de tiempo con temporizadores. Todo ello con el uso mínimo de librerías de alto nivel, concretamente en lenguaje C.

Descripción

El juego consiste en un sobre que se mueve a la izquierda y a la derecha mediante la cruceta de la nintendo, se trata de obtener el mayor número de billetes posibles en un determinado tiempo.

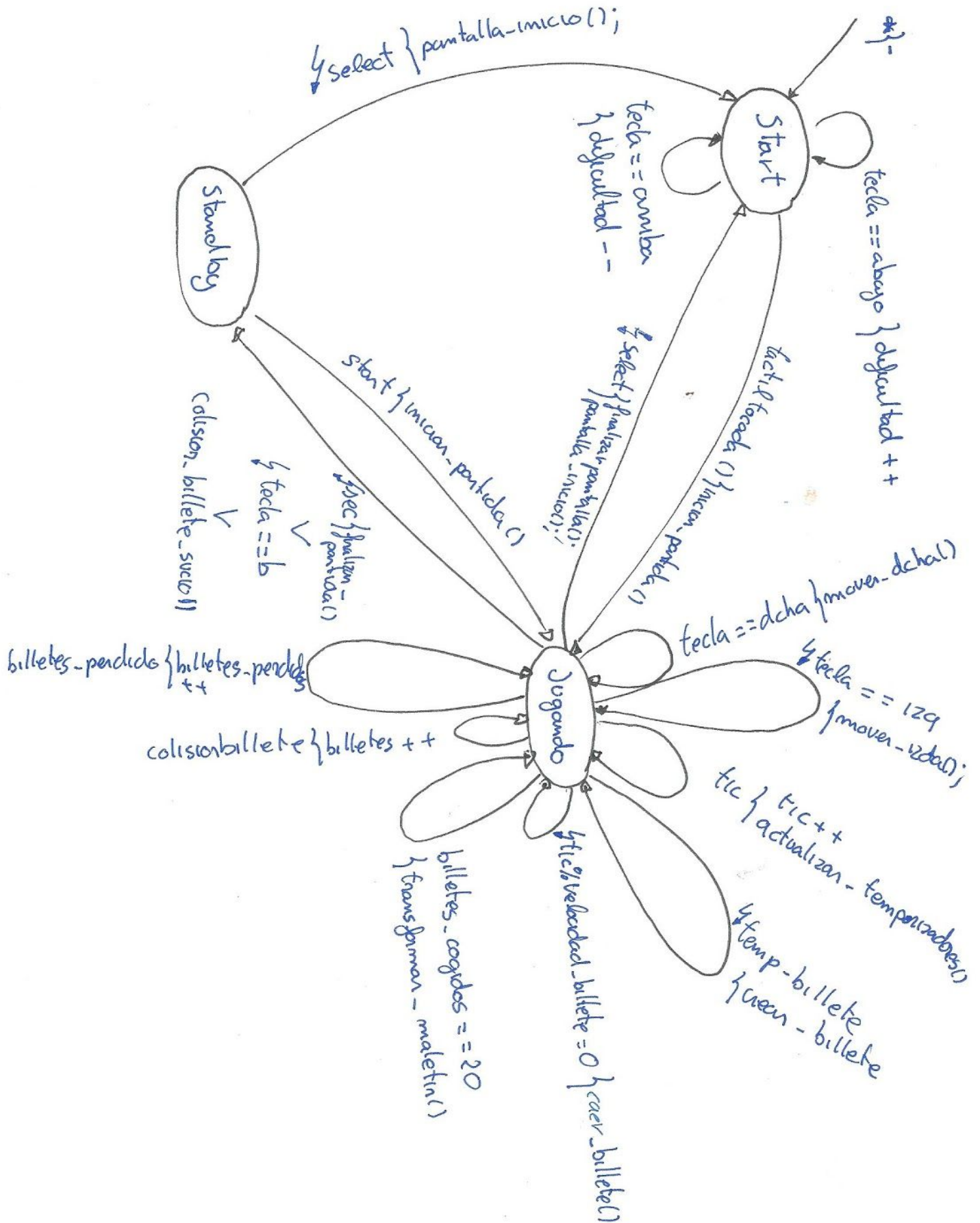
Al iniciar el juego aparece en la pantalla superior la opción de elegir la dificultad de la partida. Cuanto mayor sea la dificultad mayor será la velocidad en la que caen los billetes. Para comenzar la partida se ha de tocar la pantalla táctil. Aparecerá el sobre y comenzaran a caer billetes verdes. Estos billetes son los que han de ser cogidos, ya que de vez en cuando aparecerán unos billetes sucios de color rojo. Si se coge uno de estos billetes se finalizará la partida, esto también sucederá si se acaba el tiempo o si en mitad de la partida se pulsa la “B”.

Mientras se está jugando se indicará en la pantalla superior la puntuación y el tiempo restante. Si se recogen cierto número de billetes el sobre se convertirá en un maletín .

Al final la partida aparecerá un menú que indica que si se pulsa el botón “START” se reiniciará la partida. En cambio, si se pulsa “SELECT” se volverá a la pantalla de inicio.



Autómata



Desarrollo

La lógica de este proyecto está separada en mayormente en 3 bloques:

- Bloque “main”: Es donde se hacen las peticiones de E/S por encuesta
- Bloque de interrupción por teclado
- Bloque de interrupción por temporizador

Para seguir de manera fiel el diseño del autómata, en cada uno de esos bloques hay un switch con la siguiente estructura:

```
switch(estado){  
    case E_START:  
    case E_JUGANDO:  
    case E_STANDBY:  
    case E_FIN:  
    default:  
        break;  
}
```

Teclado

Encuesta	Interrupción
Flecha derecha	Flecha izquierda
Start	SELECT
Pantalla táctil	B

Para habilitar las interrupciones por teclado, los bits activados en el controlador del teclado son los siguientes:

- 1: B
- 2: SELECT
- 5: Flecha izquierda
- 14: Habilitar interrupciones por teclado

Temporizador

El temporizador TIMER0 está configurado para que funcione a la frecuencia máxima de 33554432 hz

TIMER0_CNT = 0xC0 (**x** = 0 + overflow habilitado + activo + ~~serie~~)

TIMER0_DAT = 0 (**latch**)

$$latch = 65.536 - \frac{1}{tics/seg} * \frac{33.554.432}{x}$$

De manera que 1 segundo = 512 tics

Flags

La función principal de los flags es agrupar todas las operaciones costosas en un solo lugar, de manera que en la sincronización por encuesta/interrupción únicamente se realizan operaciones esenciales como cambiar el valor de una variable (sin realizar bucles ni imprimir por pantalla).

Los flags, es un dato tipo **int**, y cada bit de ese int tiene asignado una tarea:

4	3	2	1	0
Sobre o maletín	Refrescar pantalla	Crear billete	Caer billete	Actualizar sobre

- **Actualizar sobre:** Imprime el sobre en la posición de pantalla indicada previamente
- **Caer billete:** Hace caer todos los billetes y detecta colisiones
- **Crear billete:** Crea un billete en una posición aleatoria de la pantalla
- **Refrescar pantalla:** Actualiza la pantalla superior
- **Sobre o maletín:** Indica si el sprite que se tiene que imprimir en pantalla es el de un sobre o un maletín

Billetes

El máximo número de billetes que se pueden crear es 10 y de cada billete guardamos 4 datos:

billete[0] : Índice (Necesario para [MostrarBillete](#) y [BorrarBillete](#))

billete[1] : Posición x

billete[2] : Posición y

billete[3] : Tipo (0:normal, 1: sucio)

Por defecto, todos los valores del billete son -1, cuando se crea un billete, se le asigna un índice, coordenadas y tipo.

Recursos utilizados



Conclusión

A modo de conclusión, hemos conseguido no solo crear un juego para la Nintendo DS en aproximadamente 13 horas (Diseño de la solución: 1h 30min, Programación: 9h, Desarrollo: 1h 30 min, Pruebas: 1h), sino que además hemos adquirido unos conocimientos más amplios en el estudio de subsistemas de entrada/salida de un computador, como en interfaces entre el exterior y la máquina. A su vez, para poder realizar el juego hemos tenido que aprender por encima como funciona el lenguaje C, el cual está desarrollado el juego.