Librería sprUtilCPC 3.0 para ccz80

Esta librería es un conjunto de funciones para el lenguaje ccz80 para Amstrad CPC que permiten gestionar sprites con unas pocas y sencillas instrucciones.

Para utilizarla en cualquier programa ccz80 es necesario utilizar la sentencia

include "sprUtilCPC.ccz80";

al principio del código, teniendo el archivo sprUtilCPC.ccz80 en la misma carpeta que el archivo del programa que se está escribiendo o en en una carpeta indicada en la opción /include del compilador.

Las funciones gestionan los sprites en cualquier modo de pantalla (0, 1 ó 2), trabajando en un sistema de coordenadas de tamaño 160 x 200 en modo 0, de 320 x 200 en modo 1 y de 640 x 200 en modo 2. La coordenada superior izquierda es la posición 0, 0.

Las imágenes que se asociarán a los sprites y las que se dibujan con sprScenery pueden ser diseñadas con la utilidad ImgEditSprUtilCPC.

Se puden gestionar hasta 25 sprites, cada uno identificado con un número desde 0 a 24. Este valor se puede moficiar cambiándolo en el código fuente de la librería modificando el valor del símbolo __sprNumber, teniendo en cuenta que un valor mayor ralentizará el programa y un valor menor lo acelerará ligeramente, además de otras consecuencias imprevisibles. Si se modifica este valor se aconseja probar diferentes hasta conseguir que el programa funcione correctamente.

El sistema genérico para realizar un juego con esta librería y las funciones asociadas a cada paso son:

1. Inicializar el sistema de sprites con sprInit.
2. Dibujar la pantalla: establecer modo, colores, crear escenario, etc. utilizando sprScenery y las funciones de la librería CPC para gestión de pantalla.
3. Activar los sprites necesarios inicialmente (se pueden activar también durante el desarrollo del juego) con sprOn.
4. Actualizar cada sprite, comprobando el valor que devuelve la función sprUpdate para verificar si se ha producido una salida de pantalla o colisión con otro sprite, y consultar la tabla de colisiones/salidas de pantalla cuya dirección devuelve sprCollisions, realizando la acción que corresponda: mover el sprite a otra posición con sprMove, sprXMove o sprYMove, cambiar la dirección del movimiento con sprShift, sprXShift, sprYShift, cambiar la imagen del sprite con sprImage, eliminar el sprite de pantalla con sprOff, etc.
5. Detectar teclas pulsadas por el/los jugador/es y modificar los desplazamientos de cada sprite controlado por el teclado o joystick mediante sprShift, sprXShift o sprYShift, y tras ello actualizar la posición del sprite, actuando como se indica para otros sprites en el punto anterior. Este paso y el del punto anterior pueden intercambiar el orden.
6. Repetir desde el punto 4.

Descripción de las funciones

sprInit(m)

Inicializa el sistema de sprites para el modo de pantalla m. Sólo se debe usar una vez al principio del programa.

sprScenery(img, x, y)

Dibuja en pantalla en las coordendas x, y la imagen img. Esta imagen no constituye un sprite, simplemente es para construir el escenario de pantalla, y por tanto no se detectarán colisiones con ella.

sprStatus(spr)

Devuelve 0 si el sprite spr no está activo y 1 si lo está.

sprOn(spr, img, x, y, x-shift, y-shift)

Activa el sprite spr dibujándolo con la imagen img y posicionándolo en las coordenadas x, y, estableciendo un desplazamiento x-shift en horizontal e y-shift en vertical para los siguientes movimientos (cada desplazamiento puede ser positivo, negativo o cero).

sprUpdate(spr)

Mueve el sprite spr a la siguiente posición, según el desplazamiento definido con sprOn o posteriormente con sprShift, sprXShift o sprYShift.

sprShift(spr, x-shift, y-shift)

Establece para próximos movimientos el desplazamiento del sprite spr en x-shift posiciones en horizontal e y-shift posiciones en vertical (cada desplazamiento pueden ser positivo, negativo o cero).

sprXShift(spr, x-shift)

Establece para próximos movimientos el desplazamiento horizontal del sprite spr en x-shift posiciones (puede ser positivo, negativo o cero).

sprYShift(spr, y-shift)

Establece para próximos movimientos el desplazamiento vertical del sprite spr en y-shift posiciones (puede ser positivo, negativo o cero).

sprMove(spr, x, y)

Establece la posición del sprite spr en las coordenadas x, y.

sprXMove(spr, x)

Establece la posición del sprite spr en la coordenada horizontal x, mientras que la coordenada vertical se mantiene igual.

sprYMove(spr, y)

Establece la posición del sprite spr en la coordenada vertical y, mientras que la coordenada horizontal se mantiene igual.

sprImage(spr, img)

Establece para el sprite spr la imagen img. El tamaño de la nueva imagen debe ser igual al de la anterior.

sprCollisions(spr, img)

Devuelve la dirección del array que contiene los valores producidos en la detección de colisiones y salidas de pantalla.

sprGetImage(spr)

Devuelve dirección de la imagen asociada actualmente al sprite spr.

sprGetX(spr)

Devuelve la coordenada horizontal actual del sprite spr.

sprGetY(spr)

Devuelve la coordenada vertical actual del sprite spr.

sprGetXShift(spr)

Devuelve el valor del desplazamiento horizontal actual del sprite spr.

sprGetYShift(spr)

Devuelve el valor del desplazamiento vertical actual del sprite spr.

sprOff(spr)

Desactiva el sprite spr.

sprFix()

Fija último sprite modificado tras actualizar su posición o imagen

sprUpdate()

Hace una pausa hasta el siguiente refresco de la pantalla.

Notas generales

- Los desplazamientos x-shift e y-shift que admiten las funciones sprOn, sprShift, sprXShift y sprYShift y los valores que devuelven sprGetXShift y sprGetYShift se consideran en complemente a dos para admitir que sean positivos o negativos, siendo su rango de -128 a +127.

- Las funciones que modifican la posición de un sprite sprOn, sprUpdate, sprMove, sprXMove y sprYMove devuelven un valor 1 indicando que el sprite colisiona con otro o está fuera de pantalla, o bien un valor 0 si la posición es correcta.

- Las funciones mencionadas en el punto anterior, cuando devuelven un valor 1 correspondiente a un error en la posición, no modifican la posición del sprite (o no lo activa si se trata de sprOn).

- Las funciones sprFix y sprUpdate se debe utilizar esta función si se observan problemas en el movimiento de los sprites. Si ello ocurre se puede probar a llamar a estas fuciones tras la utilización de las funciones que dibujan o borran sprites (sprOn, sprUpdate, sprImage, sprMove, sprXMove, sprYMove y sprOff), bien tras cualquiera de esas llamadas o sólo tras alguna de ellas, hasta conseguir que el movimiento de los sprites sea correcto.