

# Arquitectura y Entornos de desarrollo para videoconsolas

## Práctica 3 Entrada y salida de datos utilizando el texto

La práctica se podrá realizar en entorno GNU/Linux en el laboratorio. Vamos a abordar el acceso a la entrada y salida de información en formato texto de la videoconsola, abordando cuestiones relativas a: el acceso a los controles o pantalla táctil para realizar la entrada de datos y el uso de las dos pantallas como terminales de texto o el almacenamiento de información para guardar los datos generados.

Recuerde que ha de guardar los resultados de sus acciones a lo largo de las prácticas para confeccionar el porfolio de prácticas, que es la forma en que evaluará el trabajo tareas y las prácticas de la asignatura. No descuide pues hacer copias de lo que necesite del laboratorio en su espacio del servidor de la asignatura.

## 1 Introducción

Trabajaremos a partir de unos ejemplos de código disponibles en la documentación del SDK; *Libnds Documentation* [1], de la que tendrá copia en local, si ha seguido lo descrito al respecto en la práctica anterior, en `${DEVKITPRO}/documentacio`. También tendrá una copia local de los ejemplos de *libnds* en el directorio `${DEVKITPRO}/examples/`. Compruébelo con las órdenes:

```
$ ls -lrth ${DEVKITPRO}/documentacio
```

```
$ ls -lrth ${DEVKITPRO}/examples
```

En esta sesión, abordaremos el uso de los controles físicos que dispone la videoconsola que hemos escogido como plataforma de pruebas (la NDS en nuestro caso). Experimentaremos con la funcionalidad de la videoconsola para configurar las pantallas en modo texto (consola) y mostrar información sobre ellas como terminales de texto de 32x24<sup>1</sup> caracteres (*tiles*) de resolución. Cerrando este capítulo con una revisión a los sistemas de ficheros disponibles que nos permiten guardar (almacenar en memoria secundaria) información.

Aprovechando que estamos estudiando el texto como medio, el boletín se complementa con dos breves anexos que recogen la propuesta de una plantilla de página web (en HTML5), así como las cuestiones propias de la configuración de las cuentas de usuario en el servidor del laboratorio para la publicación de los contenidos web que se desarrollen.

## 2 Gestión de los controles de la videoconsola NDS

En la práctica anterior se experimentaba sobre parte de los contenidos del apartado *User Input/output* de la documentación de *libnds* [1]. a partir del estudio de los ejemplos disponibles en el directorio `${DEVKITPRO}/examples/input/`. En aquel caso, se abordó el uso de la pantalla táctil mediante el dedo o el puntero (*stylus*), averiguar su posición y área de contacto, así como la opción de utilizar un teclado en pantalla que permita introducir texto.

Ahora completamos el estudio de la entrada de datos del usuario, aproximándonos al uso del teclado físico de la videoconsola. En esta encontramos trece botones:

- Diez accesibles en la parte delantera de la base de la videoconsola. Esto es los cuatro de la cruceta de flechas de dirección, *X*, *Y*, *A*, *B*, *start* y *select*.
- Dos gatillos en la parte trasera de la base, izquierdo y derecho, habitualmente como denominados *L* y *R*, por su posición a un lado u otro de la plataforma.

---

1 Véase la documentación al respecto de *console.h*, p. ej. en <http://libnds.devkitpro.org/a00107.html>

- Y el “botón” de detección de tapa abierta o cerrada, en la bisagra de la consola..

**Ejercicio:** Implemente una aplicación que muestre el estado de los trece controles de la NDS. Para ello, a partir del ejemplo de código del listado 1 y del listado 2. Realice una versión, añadiendo al código original, la función que implementa: la lógica de detección del estado de los botones de la consola, completando el listado de botones con los valores del enumerado *KEYPAD\_BITS* definido en *input.h*<sup>2</sup> y los muestre en pantalla.

```
#include <nds.h>
#include <stdio.h>

int main(void)
{
    int held;

    consoleDemoInit();
    while(1)
    {
        scanKeys();
        held = keysHeld();

        if( held & KEY_A)
            printf("Key A is pressed\n");
        else
            printf("Key A is released\n");

        if( held & KEY_X)
            printf("Key X is pressed\n");
        else
            printf("Key X is released\n");

        if( held & KEY_TOUCH)
            printf("Touch pad is touched\n");
        else
            printf("Touch pad is not touched\n");

        swiWaitForVBlank();
        consoleClear();
    }
}
return 0;
}
```

*Listado 1: Ejemplo de código [2] para la detección del estado de algunos controles de la NDS.*

---

<sup>2</sup> Accesible también en

<<http://libnds.devkitpro.org/a00123.html#aa27cad8fa018a58930b6622783a83072>>

```

int teclaA=0;
int teclaB=0;
int teclaL=0;
int pantallaC=0;
int pantallaA=0;

void logica(){
    scanKeys(); //actualizamos los botones pulsadas

    if (keysHeld()) iprintf("\x1b[3;1HAlgun boton esta pulsado "); //miramos
    si alguna tecla está pulsada
    else iprintf("\x1b[3;1HNINGun boton esta pulsado");

    if (KEY_A & keysDownRepeat()) teclaA++; //si la tecla A está presionada
    (mediante repeticiones) se incrementa

    if (KEY_B & keysDown()) teclaB++; //cuenta las veces que se presiona B

    if (KEY_LID & keysDown()) pantallaC++; //miramos si cerramos la táctil
    if (KEY_LID & keysUp()) pantallaA++; //miramos si cerramos la táctil

    if (KEY_L & keysUp()) teclaL++; //cuenta las veces que se presiona B

    if (KEY_TOUCH & keysHeld()) { //si hay presión sobre la táctil entonces
    miramos las coordenadas. Para una selección quizás mejor keysDown
        touchPosition touch=touchReadXY();
        iprintf("\x1b[1;1HTactil X %d, Y %d ",touch.px, touch.py);
    }

    iprintf("\x1b[4;1HEl boton A se ha pulsado %d",teclaA);
    iprintf("\x1b[5;1HEl boton B se ha pulsado %d",teclaB);
    iprintf("\x1b[6;1HEl boton L se ha levantado %d",teclaL);
    iprintf("\x1b[8;1HPantalla cerrada %d",pantallaC);
    iprintf("\x1b[9;1HPantalla abierta %d",pantallaA);

};

```

*Listado 2: Ejemplo de código [3] para la detección de los controles de la NDS.*

### 3 Uso avanzado del texto en pantalla

Hemos revisado ya el uso de las secuencias de escape para ampliar el control que tenemos sobre la visualización en pantalla del texto con colores o posicionamiento del mismo a partir del ejemplo en el subdirectorio `${DEVKITPRO}/examples/Graphics/Printing/ansi_console/`.

También se ha propuesto obtener un esqueleto de código para “escribir texto” en ambas pantallas a partir de `${DEVKITPRO}/examples/Graphics/Printing/print_both_screens/s` funcionalidad..

Veremos más adelante que el uso de la aceleración hardware para el tratamiento de los gráficos en la videoconsola hace posible operaciones muy efectivas. Ahora solo nos detenemos a anotar referencias a estas operaciones.

**Ejercicio:** Tome capturas de pantalla y anote la funcionalidad de los ejemplos siguientes:

- `Graphics/Printing/custom_font/`
- `Graphics/Printing/print_both_screens`

- Graphics/Backgrounds/rotation
- Graphics/Printing/rotscale\_text/
- Graphics/Printing/console\_windows/

## 4 Sistemas de archivos en NDS

Es posible acceder a los ficheros y directorios de una manera muy similar a cómo se realiza en un computador . Para comprobarlo vamos a revisar la parte del API de *libnds* de acceso al sistema de archivos de la tarjeta de memoria que incorpora el cartucho

**Ejercicio:** Anote en una tabla las funciones propias de cada tipo de sistema de ficheros (*nitrofs* y *fat*) que se observan en los ejemplos que encontrará dentro de los subdirectorios *.filesystem/libfat/* y *filesystem/nitrofs.en* `${DEVKITPRO}/examples/templates/`. Anote en otra tabla, el resultado de la ejecución de ambos ejemplos en los dos emuladores disponibles y en la máquina real. En el emulador DeSmuME, la fig 1 muestra el resultado de *nitrodir.nds* y *libfatdir.nds*.-

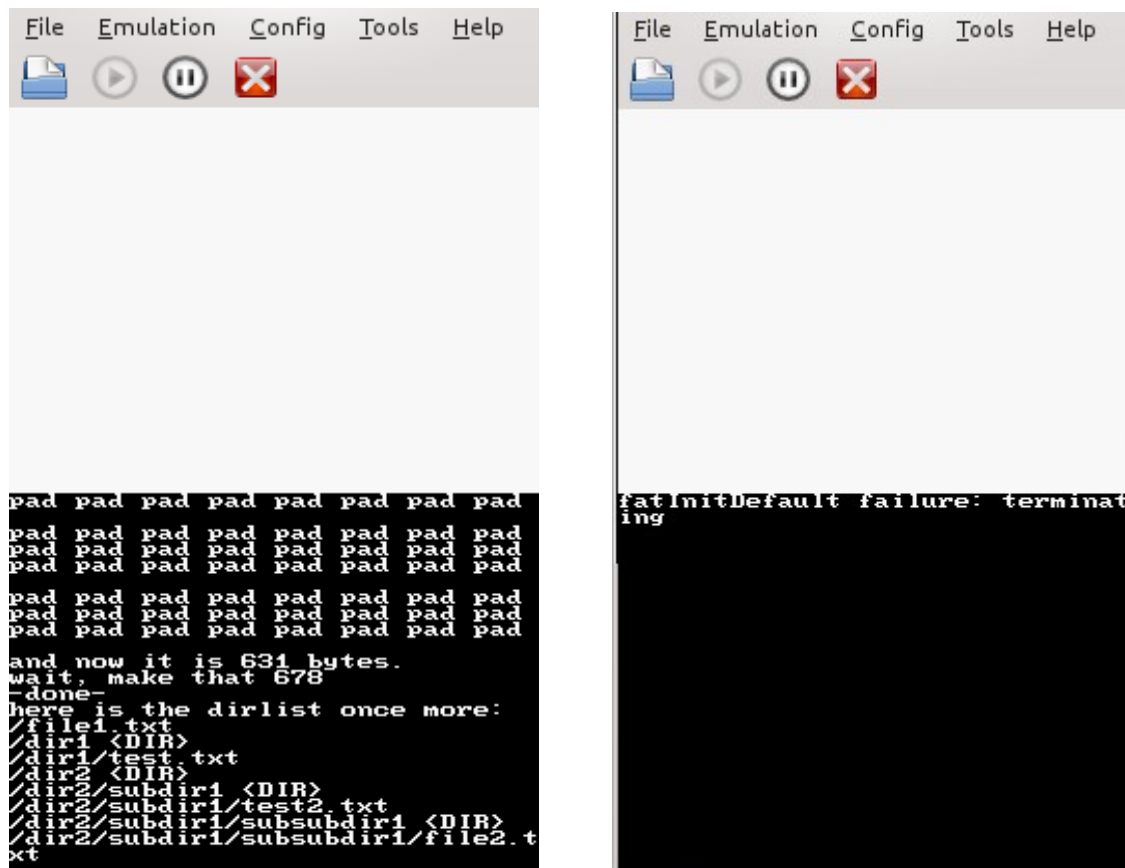


Figura 1: Resultado de la ejecución de *nitrofs* (izquierda) y *libfat* (derecha) en DeSmuME.

## 5 Trabajo autónomo

Tome la plantilla de código HTML que se ha descrito en clase y que se resume en el y publíquela incluyendo los datos que dispone en este momento sobre el trabajo de asignatura que está realizando. Los contenidos irán redefiniéndose y ampliándose conforme avance el trabajo, así que no se preocupe ahora por la extensión.

## 6 Bibliografía

- [1] Libnds Documentation, Disponible en la URL <<http://libnds.devkitpro.org/index.html>>.
- [2] NDS/Tutorials Day 2 [en línea]. Dev-Scene NDS Tutorials. 2013 [fecha de consulta: 19 de febrero del 2014]. Disponible en <<http://www.dev-scene.com/NDS>>.
- [3] [DS] DS CODE 1: "Interrumpiendo". [fecha de consulta: 19 de febrero del 2014]. Disponible en <[http://www.elotrolado.net/hilo\\_ds-ds-code-1-quot-interrumpiendo-quot\\_579855#p1704954093](http://www.elotrolado.net/hilo_ds-ds-code-1-quot-interrumpiendo-quot_579855#p1704954093)>.

## 7 Anexo A. Plantilla de página web HTML5

Esta plantilla es una propuesta de una página de contenido y su simplificada estructura de navegación. Así, se ofrece un esquema básico de publicación de los trabajos de la asignatura [AEV](#) formado por cuatro elementos:

- El "inicio", que acoge tema de trabajo, los nombres de pila y direcciones de correo de los integrantes de un grupo, la asignatura y el curso en que se realiza el trabajo. La información de la portada/contraportada que acompañaría al CD en que se pudiera distribuir también tiene cabida aquí.
- El "blog", este mismo documento, donde se puede relatar las etapas del desarrollo.
- La "galería", donde se puede recoger el desglose de los media empleados.
- El "acerca de", es el lugar donde poner las referencias a las fuentes que se hayan utilizado en la elaboración del trabajos.

Con este material se propone una rápida "inmersión" en este estándar para publicar documentos en la web que no permitan incluir objetos de naturaleza multimedia para describir un trabajo de asignatura. Estos objetos, además del usual texto e imágenes en mapas de bits, ahora toman carta de naturaleza propia el audio y el vídeo. Se puede consultar más información al respecto de este tema en [HTML5: W3C Working Draft 11 October 2012](#).

En el apartado de [Audio y vídeo en HTML5](#) se contienen ejemplos del uso de estos nuevos objetos en una página web de acuerdo a las recomendaciones del estándar HTML5, el cual se encuentra en fase final de su redacción en el momento de redactar este texto.

## 8 Anexo B. Acceso a los servidores de la asignatura y publicación de documentos

En *web-sisop.disca.upv.es* los alumnos de la asignatura pueden publicar documentos que son

accedidos desde la web. Así, por ejemplo una persona con usuario "usuario1" podrá acceder a una URL como `<http://web-sisop.disca.upv.es/~usuario1/>` y obtendrá como resultado el contenido por efecto: el de un fichero de nombre "index.html".

Para subir ficheros a *web-sisop.disca.upv.es*, se hace a través de *shellabs.disca.upv.es*<sup>3</sup>, si se está trabajando fuera del laboratorio o si se ha iniciado sesión en MS/Windows. Si se está trabajando en GNU/Linux, ya se está accediendo al espacio personal que es compartido con las dos máquinas a la vez, pero que ofrecen servicios diferentes sobre las mismas cuentas: SSH con *shellabs* y HTTP con *web-sisop*.

Para crear este y otros ficheros (y directorios) podéis acceder a un espacio de usuario (con el nombre y la contraseña que decidisteis en la matricula) al que se puede acceder con un protocolo seguro (por ejemplo *putty*, *Winscp* o *Secure Shell*) en la máquina *shell-sisop.disca.upv.es*.

Para poder publicar documentos en la red deberéis comprobar que existe un directorio "public\_html" en el directorio raíz del usuario. Si a este directorio y a los ficheros que vayáis subiendo allí les dais los permisos correspondientes tendréis inmediatamente una página web en el servidor de la asignatura desde el que se pueda ver el trabajo que realicéis en la misma.

Por ejemplo si se crea allí un fichero "index.html" y desde un terminal se comprueban los permisos de los ficheros allí contenidos se puede ver algo (salvando las diferencias de fechas, tamaños e identificación del propietario y grupo) como:

```
$ ls -la
drwxr-xr-x  18 usuario1      grup          3952 2004-10-07 19:16 .
drwxr-xr-x  30 usuario1      grup          2048 2004-10-07 19:16 ..
rw-r--r--   1 usuario1      grup           6783 2004-10-07 19:15 index.html
```

De no ser así, habrá que establecerlos en el modo indicado. El directorio raíz del usuario también debe ser modificado en este mismo sentido para dejar llegar a cualquier usuario a leer los ficheros del directorio "public\_html".

Comprobados estos permisos y fijados los cambios pertinentes debe ser posible ver el contenido del fichero "index.html" con una URL como `<http://web-sisop.disca.upv.es/~usuario1/>`. Si se quiere acceder a un fichero (*f1.html*) dentro de un directorio (*dir2*) que está dentro de otro (*dir1*) que está en el *public\_html*, la URL sería `<http://web-sisop.disca.upv.es/~usuario1/dir1/dir2/f1.html>`.

---

3 Esta máquina, *shellab.disca.upv.es*, va a sustituir a la actual *shell-sisop.disca.upv.es*, que todavía da servicio y que quizá es todavía referenciada en algún boletín de prácticas. Como es cambio se realizará durante el curso 2k13/2k14 intentaremos siempre referencia a la que se va a quedar *shellab.disca.upv.es*.