

Fundamentos de Sistemas Embebidos

Proyecto: Consola de videojuegos Retro

Castillo Montes Pamela

1. Introducción

Este proyecto tiene como objetivo central poner en práctica los conocimientos teóricos y prácticos adquiridos a lo largo del curso sobre sistemas embebidos. El resultado tangible de este aprendizaje se concreta en la creación de una consola de videojuegos retro.

En el transcurso de la materia, hemos abordado la complejidad de los sistemas embebidos mediante prácticas, razón por la cual, la consola de videojuegos desarrollada incorpora las características clave de estos sistemas, permitiéndonos aplicar directamente los conceptos aprendidos.

Cabe destacar que para la implementación de este proyecto hemos optado por utilizar una Raspberry Pi, una herramienta versátil que se ajusta a las necesidades específicas de nuestro proyecto.

2. Materiales

Para llevar a cabo este proyecto, será necesario contar con los siguientes elementos:

- Una Raspberry Pi 3B o un modelo más reciente.
- Una fuente de alimentación de 5V 1A a USB-C, o en su defecto, un cubo cargador de 5V 1A junto con un cable conector que vaya desde el puerto de salida del cubo (generalmente USB o USB-C) hacia USB-C.
- Una tarjeta microSD con capacidad mínima de 8GB, cargada con el sistema operativo Raspbian OS (Legacy) with desktop .
- Acceso a Internet.
- Un monitor compatible con HDMI o VGA.
- Cable HDMI a microUSB o VGA a microUSB, según el puerto disponible en el monitor seleccionado.
- Un mouse USB con cable.

- Un teclado USB con cable.
- Un control, preferentemente del tipo XBOX ONE.
- Opcional: Altavoces o dispositivo de audio según las preferencias del usuario.
- Opcional: Un dispositivo de almacenamiento USB.

Es importante mencionar que, aunque es posible utilizar un ratón, teclado y control inalámbricos, en este documento no se proporcionan las instrucciones específicas para su configuración y posterior conexión mediante Bluetooth

3. Antecedentes

Para llevar a cabo la instalación y configuración efectiva del proyecto presentado, es fundamental poseer un conocimiento básico sobre los componentes esenciales más importantes, tales como:

Raspberry Pi Se considera a la Raspberry Pi como una computadora de bajo costo y compacta, o bien una tarjeta microcontroladora, la cual puede ser usada en conjunto con monitores, mouse y teclados para la ejecución de diversos programas en un ambiente de desarrollo computacional. [3]

Sistema operativo Raspbian Raspbian es un sistema operativo gratuito basado en Debian y optimizado para operar en conjunto con la arquitectura y hardware de Raspberry Pi. Este sistema operativo tiene alrededor de 35,000 librerías que permiten un óptimo desarrollo y utilización de la tarjeta Raspberry Pi.[4]

Emulador Snes Se trata de un software diseñado para emular el funcionamiento del hardware de la consola SNES, lo que posibilita la ejecución de los juegos de SNES en plataformas distintas, comúnmente en dispositivos modernos o personalizados.[2]

4. Advertencia de riesgos

Es crucial destacar que la implementación de este proyecto, al igual que cualquier otro proyecto, conlleva ciertos riesgos con magnitudes variables. Dentro de los cuales se destacan los siguientes, durante el proceso de ensamblaje y configuración se deben tomar medidas para evitar posibles riesgos de electrocución, especialmente al manejar la fuente de alimentación conectada a la corriente eléctrica.

Además, se aconseja evitar el contacto directo o prolongado con componentes que puedan sobre-calentarse, tales como la Raspberry Pi o el adaptador de corriente, esto con

la finalidad de prevenir posibles quemaduras.

5. Cuidado de componentes electrónicos

Asimismo, es fundamental tener cuidado al manipular componentes electrónicos delicados, como la tarjeta microSD y los conectores de la Raspberry Pi, por lo que se recomienda utilizar herramientas antiestáticas o descargarse estáticamente antes de tocar estos componentes para evitar posibles daños.

De igual forma, al extraer o insertar la tarjeta microSD, asegúrese de que la Raspberry Pi esté desconectada de la corriente eléctrica para evitar cortocircuitos accidentales. Es de suma importancia seguir las indicaciones proporcionadas por los fabricantes para evitar daños en los dispositivos y garantizar un manejo seguro de los componentes electrónicos.

6. Configuración de la tarjeta Raspberry Pi

Se parte del supuesto de que el usuario ha descargado e instalado correctamente el sistema operativo *Raspbian OS (Legacy) with desktop* en una tarjeta microSD.

Para iniciar la configuración de la Raspberry Pi, es necesario insertar la microSD en la ranura correspondiente y conectar un monitor mediante el puerto HDMI de la tarjeta, todo ello antes de alimentar la Raspberry Pi conectándola a la fuente de energía.

Cabe destacar que, para llevar a cabo la configuración de manera integral, se requiere la conexión de un teclado y un ratón USB al sistema.

Conexión a internet Dado que la descarga de archivos y bibliotecas es esencial y depende de la conexión a Internet, es necesario que el usuario configure el acceso a la red, ya sea a través de una conexión Ethernet o mediante Wifi.

Resolución de pantalla Si la pantalla no se ajusta adecuadamente a su monitor, puede realizar ajustes accediendo al menú principal, navegando a Preferences — > Screen Resolution, y seleccionando el valor que mejor se adapte a sus necesidades.

Salida de sonido El siguiente paso es configurar la salida de sonido. Para hacerlo, simplemente haga clic izquierdo sobre el icono de sonido en la barra de tareas y elija la salida de sonido deseada.

7. Librerías a utilizar

A continuación, se listan las bibliotecas necesarias junto con el comando utilizado para su instalación:

```
sudo apt install libsdl2-dev libgtkmm-3.0-dev libepoxy-dev meson alsa-oss portaudio19-  
dev libminizip-dev  
sudo apt-get install ninja-build  
sudo apt install xboxdrv  
sudo apt install joystick  
sudo apt-get install fbi  
sudo apt-get install python3-pil python3-pil.imagetk
```

8. Instalación automática

Es necesario descargar el proyecto desde el repositorio de GitHub. Para obtener una explicación detallada sobre cómo realizar este procedimiento, siga el Paso 1 descrito en la sección de 'Instalación manual'.

Una vez completado el paso anterior, ejecute el siguiente comando desde la terminal.

```
chmod +x /home/pi/RetroRaspy/Proyect.sh  
sh /home/pi/RetroRaspy/Proyect.sh
```

Finalmente, espere a que el proceso termine y verifique su funcionamiento. Continúe con la instalación a partir del Paso 6 en la sección de 'Instalación manual'.

9. Instalación manual

En esta sección, se describen los pasos a seguir para una correcta instalación manual de todos los componentes del proyecto, así como su posterior ejecución.

9.1. Paso 1: Descarga de proyecto

La descarga de los archivos necesarios para la implementación del proyecto se lleva a cabo mediante diversos métodos que permiten clonar el repositorio. A modo de ejemplo, se pueden emplear los siguientes comandos en la terminal para obtener los documentos requeridos.

```
cd /home/pi  
wget https://github.com/colpim2/RetroRaspy
```

Cabe destacar que el proyecto está configurado para operar dentro de la ruta mencionada. En caso de desear utilizar una ruta diferente, es de suma importancia modificar todas las rutas, tanto relativas como absolutas, dentro del proyecto para que coincidan con la nueva ubicación deseada.

9.2. Paso 2: Descarga y compilación del emulador

Continuando con el proceso, es esencial descargar y compilar el emulador snes9x [1], y en este caso, usaremos la versión 1.60gtk. Para lograrlo, ejecutaremos los siguientes comandos en la terminal.

```
cd /home/pi/RetroRaspy/Emulador
wget https://github.com/snes9xgit/snes9x/archive/refs/tags/1.60.tar.gz
tar -xzf 1.60.tar.gz
cd snes9x-1.60/gtk
meson build --buildtype=release --strip
```

Tras finalizar el proceso de compilación, podemos verificar la instalación correcta ejecutando los siguientes comandos en la terminal.

```
cd /home/pi/RetroRaspy/Emulador/snes9x-1.60/gtk/build
./snes9x-gtk
```

9.3. Paso 3: Configuración del gamepad/control/teclado

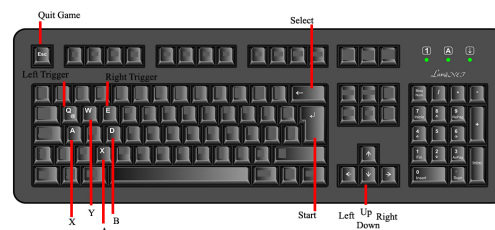
Como se mencionó en el listado de materiales, se utilizará un control de XBOX ONE conectado mediante USB, así como la opción de un teclado para el control dentro del emulador. Procedemos a conectar el control a la Raspberry Pi y encenderlo antes de ejecutar el emulador.

Archivo *snes9x.config* Dentro de los archivos descargados del programa, se incluye el archivo con el mapeo del control y teclado. Para copiar este archivo en el directorio oculto .config, utilice los siguientes comandos. Si desea visualizarlo en el explorador de archivos, haga clic izquierdo dentro del explorador de archivos — > Mostrar archivos ocultos.

```
sudo cp /home/pi/RetroRaspy/snes9x.conf /home/pi/.config/snes9x/
```



(a) XBOX ONE



(b) Teclado

Figura 1: Mapeado

9.4. Paso 4: Gestión de ROMS

Dentro de los archivos del proyecto descargados se encuentran 15 ROMs ubicadas en la carpeta ../Emulador/Roms. Si se desea agregar más, se pueden utilizar dos métodos distintos.

9.4.1. Descarga de ROMs

Descargue las ROMs deseadas y guárdelas manualmente en la carpeta Roms, que se encuentra en ../Emulador/Roms. Asegúrese de que el archivo cumpla con las siguientes características:

- Evite el uso de caracteres especiales en el nombre del archivo.
- Asegúrese de que el archivo tenga la extensión *.sfc* o *.srm*.

Nota: Recuerde descargar las ROMs desde sitios seguros y asegurarse de tener los derechos correspondientes para evitar la infracción de derechos de autor.

9.4.2. Gestión de ROMs desde USB

Este segundo método es funcional una vez ejecutada la interfaz del siguiente paso. Para implementarlo, simplemente ejecute la interfaz y conecte la memoria USB a la Raspberry Pi. El sistema se encargará automáticamente de buscar archivos que cumplan con las siguientes características y copiarlos al directorio Roms.

- Evite el uso de caracteres especiales en el nombre del archivo.
- Asegúrese de que el archivo tenga la extensión *.sfc* o *.srm*.

Para garantizar el buen funcionamiento de esta opción, se recomienda que los archivos se encuentren preferiblemente en el directorio raíz de la USB. Además, es importante otorgar tiempo suficiente para la carga y copia de archivos. La finalización de este proceso se indica mediante la actualización de la lista de ROMs disponibles para jugar.

El tiempo de ejecución de este proceso oscila entre los 20 segundos después de insertar la USB hasta 1 minuto, dependiendo de la cantidad de archivos presentes en la USB y la cantidad de archivos a copiar.

9.5. Paso 5: Interfaz

Dentro de los archivos descargados, se incluye un archivo llamado 'Interfaz.py', que contiene el código utilizado para la interfaz del proyecto. En esta interfaz, se presenta inicialmente un listado de las ROMs disponibles. Al hacer clic en alguna de ellas, se selecciona y puede ejecutarse presionando el botón 'Jugar'. Esto cargará automáticamente el emulador con el juego seleccionado.

En caso de que no desee que la tarjeta Raspberry se dedique exclusivamente a la ejecución del emulador, es posible ejecutar esta interfaz desde la terminal mediante el siguiente comando o bien ejecutar el programa con su IDE de preferencia:

```
python3 /home/pi/RetroRaspy/Emulador/Interfaz.py
```

9.6. Paso 6: Reproducción de imagen estática en el arranque

Para la configuración de la imagen estática durante el inicio o arranque de la tarjeta, se deben de seguir los siguientes pasos

Archivo */boot/config.txt*

Dentro de la terminal, ejecutar

```
sudo nano /boot/config.txt
```

Agregue al final del archivo la línea *disable_splash=1*. Guarde y salga del archivo con *Ctrl+O* y *Ctrl+X*.

Archivo */boot/cmdline.txt*

Dentro de la terminal, ejecutar

```
sudo nano /boot/cmdline.txt
```

Modifique *console=tty1* por *console=tty3* y agregue al final de la línea: *splash loglevel=3 logo.nologo vt.global_cursor_default=0 plymouth.enable=0*. Guarde y salga del archivo con *Ctrl+O* y *Ctrl+X*.

Asegúrese de que el contenido del archivo esté escrito en una línea continua y esta sea la primera línea del documento para evitar posibles errores.

Activación del Servicio *splashscreen*

Copie el archivo *splashscreen.service* y active el servicio con los siguientes comandos:

```
sudo cp /home/pi/RetroRaspy/splashscreen.service /etc/systemd/system/  
sudo systemctl enable splashscreen
```

9.7. Paso 7: Arranque directo al emulador

Por último, compruebe que todos los pasos anteriores funcionen correctamente ya que una vez realizado este paso, no podrá acceder al escritorio para realizar cambios.

En la terminal ejecuta los siguientes comandos para crear las carpetas */lxsession/LXDE-pi* y copiar el archivo *autostart* del sistema operativo a estas.

```
mkdir -p /home/pi/.config/lxsession/LXDE-pi
cp /etc/xdg/lxsession/LXDE-pi/autostart /home/pi/.config/lxsession/LXDE-pi/
nano /home/pi/.config/lxsession/LXDE-pi/autostart
```

Una vez abierto el editor de texto, agrega al final del archivo la siguiente línea asegurandote de ajustar la ruta para que coincida con la de tu proyecto.

```
@python3 /home/pi/RetroRaspy/Emulador/Interfaz.py
```

Guarde y salga del archivo con *Ctrl+O* y *Ctrl+X*. Cierre sesión y vuelva a iniciarla o reinicie la Raspberry Pi para que los cambios surtan efecto.

10. Troubleshooting

Se listan algunos errores comunes que puede experimentar al realizar la instalación y configuración del proyecto, así como sus correspondientes soluciones

Fallo en la actualización del sistema

Asegúrese de que el sistema esté completamente actualizado utilizando el comando

```
sudo apt update && sudo apt upgrade
```

Problemas con el Sistema Operativo

Verifique que el sistema operativo esté correctamente instalado y sea compatible. Si experimenta problemas, considere reinstalar o actualizar a una versión compatible.

Falta de salida de audio en el monitor

Asegúrese de que el monitor conectado tenga salida de audio o de que los dispositivos de audio estén debidamente conectados y configurados en el sistema.

Instalación incorrecta o incompleta de bibliotecas necesarias

Verifique que todas las bibliotecas necesarias estén instaladas correctamente. Utilice los comandos de instalación especificados para instalar la totalidad de las bibliotecas mencionadas.

Uso de Rutas Relativas en Lugar de Rutas Absolutas

Revise y modifique las rutas descritas en los archivos utilizados. Siempre utilice rutas absolutas en lugar de rutas relativas para evitar problemas de ubicación de archivos.

Configuraciones Incorrectas en el Emulador SNES

Revise y verifique las configuraciones dentro del emulador SNES para asegurarse de que estén correctamente ajustadas según los requisitos del proyecto.

Problemas con los Drivers del Control Utilizado

Asegúrese de que los controladores del control utilizado estén correctamente instalados y actualizados para garantizar su correcto funcionamiento.

Presencia de Múltiples Versiones Instaladas del Emulador

Desinstale versiones adicionales del emulador y mantenga solo la versión necesaria para evitar conflictos.

Mapeo Incorrecto de Teclas

Verifique el mapeo de teclas en el archivo oculto dentro de `.config/snes9x` para asegurarse de que esté correctamente configurado según las especificaciones de su control.

11. Conclusión

En conclusión, en este proyecto se logro la aplicación práctica de los conocimientos teóricos y prácticos adquiridos sobre sistemas embebidos. Obteniendo así la creación de una consola de videojuegos retro.

Dentro del proceso de desarrollo de el mismo, nos percatamos que a pesar de la amplia documentación existente para la tarjeta Raspberry Pi, debido a su constante desarrollo y actualización, una gran parte de esta se ha vuelto obsoleta y no es compatible con versiones actuales de los sistemas operativos. En respuesta, optamos por utilizar la versión Legacy del sistema operativo, asegurando así una mayor compatibilidad con las soluciones disponibles.

De igual forma, al limitarnos a no utilizar soluciones existentes, se incrementa la exigencia de tiempo y la necesidad de un conocimiento más detallado sobre el funcionamiento de la tarjeta para llevar a cabo el desarrollo del proyecto. Esta decisión también implica restricciones en cuanto a los dispositivos que podrían integrarse en nuestro sistema. Un ejemplo de esto es el caso de los gamepads, ya que cada uno hace uso de controladores específicos, y sin disponer de la totalidad de estos dispositivos, no es posible garantizar su compatibilidad.

Github y video

Los códigos utilizados para la realización del proyecto se encuentran en el siguiente: Repositorio de Github.

De igual forma, se anexa un link donde se puede visualizar la evidencia multimedia del funcionamiento del proyecto: Vídeo Youtube.

Referencias

- [1] bearoso (2023). snes9x. <https://github.com/snes9xgit/snes9x>. Consultado: 2023-11-27.
- [2] IBM documentation (2023). Emuladores. <https://www.ibm.com/docs/es/aix/7.2?topic=concepts-emulators>. Consultado: 2023-11-27.
- [3] Raspberry (s.f). ¿Qué es Raspberry Pi? <https://raspberrypi.cl/que-es-raspberry/>. Consultado: 2023-11-27.
- [4] Raspbian (s.f). Welcome to Raspbian. <https://www.raspbian.org/>. Consultado: 2023-11-27.