

Desarrollo de un Driver de Dispositivo de Caracteres para Control de LED vía GPIO

Lychee RV Dock - Embedded Linux System Programming 2025-1S

Diego Alejandro Arboleda Cuero

Junio de 2025

Descripción del Proyecto

Este proyecto implementa un **driver de dispositivo de caracteres para Linux** que permite controlar un LED conectado a un pin GPIO desde el espacio de usuario, usando una interfaz en `/dev/led_driver`.

El sistema está dividido en dos partes:

- **Módulo del kernel:** configura el GPIO y escucha comandos `.\n` y `.\off`.
- **Aplicación en espacio de usuario:** muestra un menú interactivo para controlar el LED desde terminal.

Objetivos de Aprendizaje

- Comprender la arquitectura de drivers tipo `character device`.
- Registrar dispositivos en el kernel y exponerlos mediante `/dev`.
- Controlar GPIO desde el kernel de Linux.
- Diseñar una aplicación de usuario que interactúe con el driver.
- Aplicar documentación, pruebas y control de versiones con Git.

Estructura del Proyecto

```
Driver_Controlador/  
src/  
    led_driver.c  
    user_app.c  
    Makefile  
Makefile  
install.sh  
test/  
README.md  
.gitignore
```

Requisitos Técnicos

Módulo del Kernel

- Loadable Kernel Module (LKM)
- Registro de dispositivo en `/dev/led_driver`
- Funciones: `init`, `exit`, `open`, `release`, `write`

- Control GPIO mediante `.on` y `.off`
- Salidas visibles con `printk()` en `dmesg`

Aplicación en Usuario

- Menú interactivo por consola
- Uso de `open()`, `write()`, `close()`

Compilación

```
make          % Compila el driver y la aplicación
make clean    % Limpia los binarios
```

Ejecución en Lychee RV Dock

Opción 1: Script Automático

```
sudo ./install.sh
```

Este script:

1. Carga el módulo
2. Obtiene el número mayor
3. Crea `/dev/led_driver`
4. Ejecuta el menú de usuario
5. Descarga el módulo al salir

Opción 2: Manual

```
sudo insmod src/led_driver.ko
sudo dmesg | tail -20
sudo mknod /dev/led_driver c <major> 0
sudo chmod 666 /dev/led_driver
./src/user_app
sudo rmmod led_driver
```

Estrategia de Pruebas

Prueba	Resultado Esperado
Carga del módulo	Se crea <code>/dev/led_driver</code>
<code>echo .on</code> al driver	LED encendido
<code>echo .off</code> al driver	LED apagado
Comando inválido (<code>echo .abc</code>)	Mensaje de advertencia en <code>dmesg</code>
Ejecución de <code>user_app</code>	Interfaz por menú funcional
Opción 0 en el menú	Descarga del módulo

Notas Importantes

- GPIO utilizado: GPIO 37
- Sistema probado en: Lychee RV Dock con Linux
- Desarrollo en: Visual Studio Code + consola serial

Criterios de Evaluación Cumplidos

- X Control de LED desde driver
- X Registro en `/dev`
- X Funciones `open`, `release`, `write`
- X Aplicación de usuario funcional
- X Evidencia de pruebas
- X Makefiles limpios
- X Documentación profesional
- X Uso progresivo de Git

Autor

- Diego Alejandro Arboleda Cuero
- Curso: Embedded Linux System Programming – 2025-1S
- Fecha: Junio de 2025