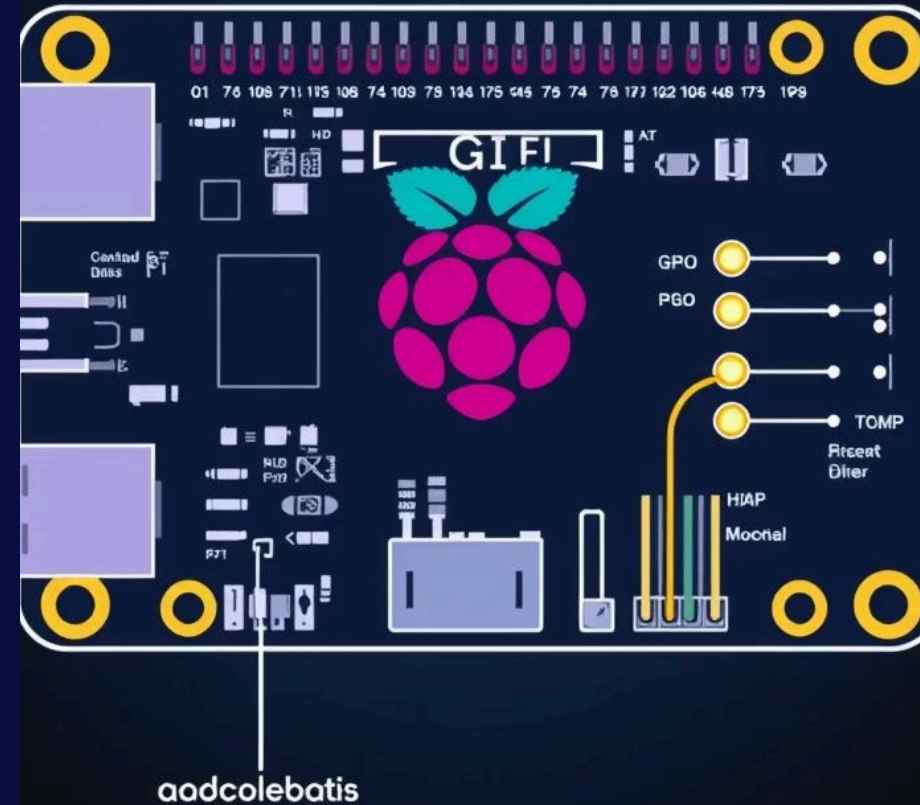


Reporte Técnico: Proyecto Final de Sistemas Operativos

Desarrollo de un Módulo de Kernel para Control de Hardware (GPIO) en Raspberry Pi.





Objetivo del Proyecto

Desarrollar un sistema modular en espacio de kernel que simule un sensor de temperatura (DHT22) y controle una matriz de LEDs 4x4, utilizando una capa de emulación de GPIOs.

Motivación y Justificación

Sistemas Embebidos

Esenciales para domótica, monitoreo ambiental e IoT, requieren comunicación precisa con sensores y actuadores.

Limitaciones Actuales

El manejo de sensores desde el espacio de usuario puede generar latencias impredecibles y limitaciones en tiempo real.

Solución Propuesta

Emulación de un controlador en espacio de kernel para mayor robustez y confiabilidad.

Alcance del Proyecto

Desarrollo y validación de un sistema modular funcional en el espacio de kernel de Linux, simulando un sistema embebido.

1

Emulación de GPIOs

Módulo gpio_sim.ko simula 32 pines digitales.

2

Simulación de Sensor

Módulo sensor_sim.ko genera temperaturas aleatorias.

3

Control de LEDs

Módulo led_matrix.ko controla una matriz de 4x4 LEDs.

4

Módulo de Control

controlador.ko lee temperaturas y actualiza los LEDs según el rango.

Conceptos Fundamentales



Espacio de Kernel

Zona protegida para ejecución de módulos y acceso a hardware.



GPIO

Pines digitales para interactuar con sensores y actuadores, simulados.



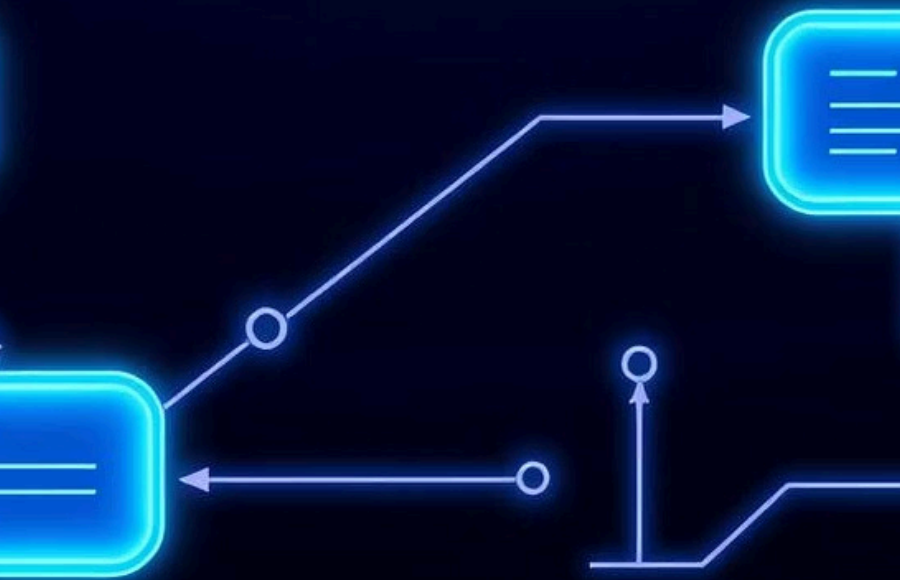
Módulos del Kernel

Componentes independientes que se comunican mediante funciones exportadas.



Kthreads y Workqueues

Hilos y colas de trabajo para monitoreo y generación de datos.



Diseño e Implementación

gpio_sim.ko

Simula 32 pines GPIO con funciones básicas de manejo.

sensor_sim.ko

Genera temperaturas aleatorias cada 5 segundos y exporta su valor.

led_matrix.ko

Controla una matriz de 4x4 LEDs usando 8 pines GPIO.

controlador.ko

Lógica principal: lee temperatura, clasifica y actualiza los LEDs.

Pruebas y Evaluación

CP-001	Temperatura < 25°C	4 LEDs encendidos	Correcto	Éxito
CP-002	Temperatura entre 25 y 29°C	8 LEDs encendidos	Correcto	Éxito
CP-003	Temperatura >= 30°C	12 LEDs encendidos	Correcto	Éxito
CP-004	Temperatura cambia de rango	Matriz actualizada una vez	Correcto	Éxito
CP-005	Temperatura cambia dentro del mismo rango	Matriz no actualizada	Correcto	Éxito

Conclusiones y Futuro

El proyecto logró simular la interacción entre un sensor de temperatura y una matriz de LEDs usando GPIOs virtuales en el kernel.

- La simulación de hardware permitió pruebas sin dispositivos físicos.
- Se fortalecieron habilidades en desarrollo a bajo nivel y manejo de estructuras del kernel.
- La IA fue clave para la depuración y documentación del código.

Accede al código y documentación en [GitHub](#)