Guía Paso a Paso del Laboratorio 03 — Módulo 03 (Video 03)

Despliegue Headless, Reglas `udev`, Control GPIO Moderno y Servicio Resiliente `systemd`

> [!NOTE]

> **Archivo Complementario Referenciado en el Video 03:** Esta guía contiene el procedimiento completo paso a paso para reproducir en laboratorio o en casa la configuración íntegra del servidor Edge Linux explicada en el **Video 03**. Todos los scripts listos para usar se encuentran en la carpeta [`codigo_fuente/`] (file:///D:/_Mecatrobotica/Video_03_Modulo_03_Servidor_Edge_Linux_Raspberry_Pi/material_complementario/codigo_fuente/).

Paso 1: Grabación *Headless* de Raspberry Pi OS Lite (64-bit) y Llaves SSH `Ed25519`

- En tu computador de ingeniería (Windows, Linux o macOS), abre una terminal y genera tu par de llaves criptográficas `Ed25519` si aún no dispones de una:

  ```bash
  ssh-keygen -t ed25519 -C "ingenieria-iot@mecatrobotica"
  ```

- Abre **Raspberry Pi Imager**, selecciona tu modelo de placa (**Raspberry Pi 4** o **Raspberry Pi 5**), elige **Raspberry Pi OS (other) ➔ Raspberry Pi OS Lite (64-bit)** (basado en Debian 12 *Bookworm*) y selecciona tu tarjeta microSD Clase A2.
- Haz clic en **Edit Settings (Personalización del SO)**:
- **Hostname:** `edge-gateway-01` (accesible en red local mediante `mDNS` como `edge-gateway-01.local`).
- **Username:** `iotadmin` y una contraseña robusta.
- **Wireless LAN / Ethernet:** Configura el SSID si usarás Wi-Fi o conecta cable Ethernet Cat 6.
- **Services (Pestaña Servicios):** Activa **Enable SSH**, selecciona **Allow public-key authentication only** y pega el contenido de tu archivo `~/.ssh/id_ed25519.pub`.
- Inserta la microSD en la Raspberry Pi, energízala con la fuente oficial de $5.1\ \text{V}$ y conéctate remotamente tras $40\ \text{segundos}$:

  ```bash
  ssh iotadmin@edge-gateway-01.local
  ```

Paso 2: Habilitación de Buses de Hardware (`I2C`, `SPI`, `UART`, `1-Wire`)

Una vez dentro de la sesión SSH en la Raspberry Pi, habilita los periféricos del kernel y libera el puerto serie de la consola de login ejecutando:

```
# 1. Habilitar bus I2C (/dev/i2c-1) y bus SPI (/dev/spidev0.0)
sudo raspi-config nonint do_i2c 0
sudo raspi-config nonint do_spi 0

# 2. Desactivar consola de login en UART (1 = desactivar consola) y activar hardware UART (0 = activar)
sudo raspi-config nonint do_serial_cons 1
sudo raspi-config nonint do_serial_hw 0

# 3. Habilitar bus 1-Wire en GPIO 4 (Pin físico 7)
echo "dtoverlay=w1-gpio,gpiopin=4" | sudo tee -a /boot/firmware/config.txt
```

Paso 3: Grupos de Privilegio Mínimo y Reglas `udev` Deterministas

- Otorga acceso directo a los dispositivos de hardware (`/dev/ttyACM`, `/dev/i2c-1`, `/dev/spidev`, `/dev/gpiochip*`) al usuario `iotadmin` sin necesidad de ejecutar tus programas como `root`:

  ```bash
  sudo usermod -aG dialout,i2c,spi,gpio iotadmin
  ```

- Copia el archivo complementario [`codigo_fuente/99-iot-arduinos.rules`] (file:///D:/_Mecatrobotica/Video_03_Modulo_03_Servidor_Edge_Linux_Raspberry_Pi/material_complementario/codigo_fuente/99-iot-arduinos.rules) a `/etc/udev/rules.d/` y recarga el gestor `udev`:

  ```bash
  sudo cp 99-iot-arduinos.rules /etc/udev/rules.d/99-iot-arduinos.rules
  ```

sudo udevadm control --reload-rules

sudo udevadm trigger

```

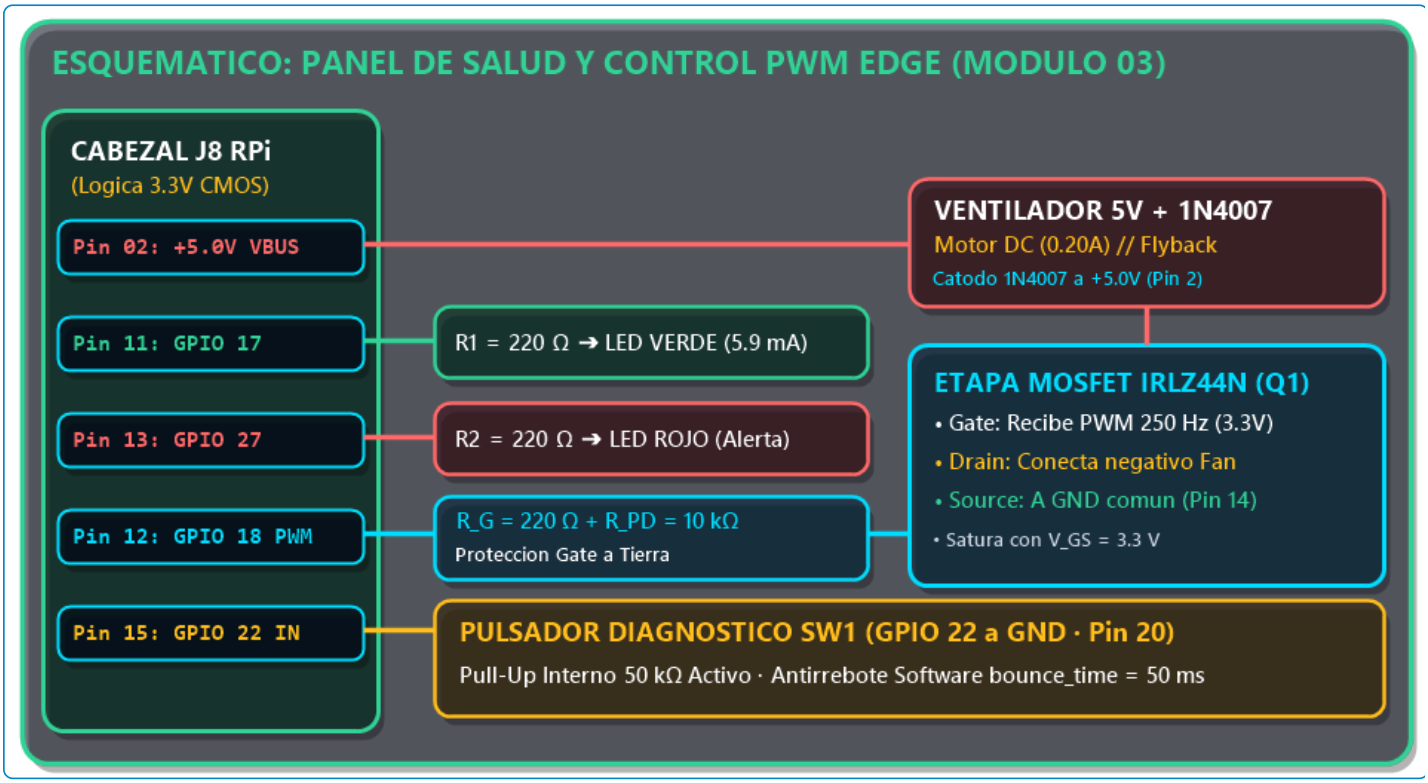
- Verifica que al conectar un Arduino UNO R3 aparezca el enlace simbólico fijo `/dev/arduino_clima`:

```bash

ls -l /dev/arduino_*

```

Paso 4: Armado del Circuito de Supervisión GPIO en Protoboard



- Conecta `GPIO 17` (**Pin físico 11**) ➜ Resistencia $220\ \Omega$ ➜ Ánodo LED Verde ➜ Cátodo a `GND` (**Pin físico 9**).
- Conecta `GPIO 27` (**Pin físico 13**) ➜ Resistencia $220\ \Omega$ ➜ Ánodo LED Rojo ➜ Cátodo a `GND` (**Pin físico 14**).
- Conecta `GPIO 18` (**Pin físico 12**) ➜ Resistencia $220\ \Omega$ ➜ Compuerta (*Gate*) del MOSFET `IRLZ44N`, añadiendo la resistencia *pull-down* de $10\ \text{k}\Omega$ entre *Gate* y `GND`.
- Conecta el ventilador de $5\ \text{V}$ entre el **Pin físico 2 ($+5.0\ \text{V}$)** y el Drenador (*Drain*) del MOSFET, colocando el diodo `1N4007` en antiparalelo (cátodo hacia $+5.0\ \text{V}$).
- Conecta el pulsador normalmente abierto `SW1` entre `GPIO 22` (**Pin físico 15**) y `GND` (**Pin físico 20**).

Paso 5: Despliegue del Demonio en Python y Servicio `systemd`

Puedes ejecutar el aprovisionamiento automático con [`codigo_fuente/provision_edge_rpi.sh`] (file:///D:/_Mecatrobotica/Video_03_Modulo_03_Servidor_Edge_Linux_Raspberry_Pi/material_complementario/codigo_fuente/provision_edge_rpi.sh) o realizar los pasos manualmente:
```

```
# 1. Instalar dependencias del sistema operativo (backend lgpio para RPi 4 y RPi 5)
sudo apt-get update
sudo apt-get install -y python3-venv python3-gpiozero python3-lgpio gpiod i2c-tools

# 2. Crear directorio de producción y entorno virtual heredando librerías C del sistema
sudo mkdir -p /opt/mecatrobotica/edge
sudo chown -R iotadmin:iotadmin /opt/mecatrobotica
python3 -m venv --system-site-packages /opt/mecatrobotica/edge/.venv

# 3. Copiar el script del demonio y otorgar permisos de ejecución
cp edge_health_daemon.py /opt/mecatrobotica/edge/edge_health_daemon.py
chmod +x /opt/mecatrobotica/edge/edge_health_daemon.py

# 4. Instalar y habilitar la unidad de servicio systemd
sudo cp edge-health.service /etc/systemd/system/edge-health.service
sudo systemctl daemon-reload
sudo systemctl enable --now edge-health.service
```

Paso 6: Pruebas de Auditoría y Resiliencia (Inyección de Fallas)

- **Inspección en Tiempo Real de `journald`:**

  ```bash
  journalctl -u edge-health.service -f
  ```

  Presiona el botón físico conectado a `GPIO 22` (Pin 15) y verifica que aparezca instantáneamente la línea `DIAGNÓSTICO MANUAL [GPIO 22] -> Temp SoC: ...`.

- **Prueba de Auto-Recuperación ante Fallo Crítico (`Crash Test`):**

  Obtén el `PID` del proceso y envíale una señal letal `SIGKILL (-9)` simulando un cuelgue abrupto:

  ```bash
  sudo kill -9 $(systemctl show --property MainPID --value edge-health.service)
  ```

  Observa en `journalctl -u edge-health.service -f` cómo `systemd` detecta la caída, espera exactamente `RestartSec=5` segundos y vuelve a levantar el servicio automáticamente restaurando el latido del LED verde.