# Guía de Integración del Módulo de Comunicación RemoteGPIO

Este documento describe la arquitectura de **RemoteGPIO** y explica cómo desarrollar un módulo de comunicación personalizado para integrar cualquier dispositivo de E/S con **Venus OS** de Victron.

## 1. Arquitectura del Sistema

RemoteGPIO se divide en dos partes para garantizar **estabilidad y modularidad**:

### 1.1 Motor Central (dbus-rgpio.py)

* Servicio persistente que no conoce protocolos de hardware específicos.
* Funciones principales:
  + Administrar el módulo del kernel para GPIO virtuales.
  + Crear y administrar servicios D-Bus (com.victronenergy.switch.\*).
  + Crear /run/io-ext/ para el descubrimiento de entradas digitales.
  + Proporcionar una **API MQTT interna** para comunicación con módulos.

### 1.2 Módulos de Comunicación (ej. dingtian\_mqtt\_bridge.py)

* Scripts independientes que se ejecutan como servicios.
* Funciones:
  + Comunicar con un dispositivo específico (MQTT, Modbus, etc.).
  + Traducir estados del hardware a llamadas de la API del Motor Central.
  + Traducir comandos del Motor Central a comandos de hardware.

## 2. Estructura de Archivos y Gestión de Servicios

### A. Ubicación de archivos

| Archivo | Ruta |
| --- | --- |
| Script de módulo | /data/RemoteGPIO/modules/ |
| Archivo de configuración .ini | /data/RemoteGPIO/conf/ |
| Servicio DaemonTools | /service/ (ej. /service/my\_new\_bridge) |

### B. Control de Servicio con setup\_rgpio.py

* Permite iniciar/detener módulos desde la GUI de Venus OS.
* Para controlar un módulo, agregarlo a /data/RemoteGPIO/conf/setup\_rgpio.ini.

**Ejemplo para Relay4:**

[device\_config]  
# ... otras configuraciones ...  
Relay4 = My New Module  
Service4 = my\_new\_bridge

* **Relay4:** nombre amigable en GUI.
* **Service4:** nombre exacto del servicio en /service/.

Cuando el usuario alterna “Relay4” en la GUI:

svc -u /service/my\_new\_bridge # iniciar  
svc -d /service/my\_new\_bridge # detener

## 3. API Interna (MQTT)

Toda comunicación con el Motor Central se realiza mediante MQTT en localhost:1883.

### A. Registro y Desregistro de Dispositivos

**Registro:**

Tema: rgpio/api/device/register/MYDEVICE\_001  
Payload: {"serial": "MYDEVICE\_001", "num\_inputs": "4", "num\_relays": "8", "device\_instance": "260"}  
Retain: true

* Crea servicio D-Bus com.victronenergy.switch.MYDEVICE\_001.
* Crea /run/io-ext/MYDEVICE\_001 con enlaces simbólicos.

**Desregistro:**

Tema: rgpio/api/device/register/MYDEVICE\_001  
Payload: "UNREGISTER"  
Retain: true

### B. Estado del Dispositivo (Conexión)

Tema: rgpio/api/device/status/MYDEVICE\_001  
Payload: "CONNECTED" # o "DISCONNECTED"  
Retain: true

* Actualiza /State y /Connected en D-Bus.

### C. Entradas Digitales (Módulo → Motor)

* Comunicación unidireccional: módulo escribe en sysfs.

input\_path = "/run/io-ext/MYDEVICE\_001/input\_2"  
new\_value = "1"  
try:  
 with open(f"{input\_path}/direction", 'w') as f:  
 f.write('out')  
 with open(f"{input\_path}/value", 'w') as f:  
 f.write(new\_value)  
 with open(f"{input\_path}/direction", 'w') as f:  
 f.write('in')  
except IOError as e:  
 logger.error(f"No se pudo escribir en la ruta sysfs: {e}")

### D. Comandos de Relé (Motor → Módulo)

Tema: rgpio/api/relay/set/MYDEVICE\_001/4  
Payload: "ON" # o "OFF"

* Módulo debe traducir a comando físico para el relé.

### E. Retroalimentación del Estado del Relé (Módulo → Motor)

Tema: rgpio/api/device/relay/read/MYDEVICE\_001/1  
Payload: "OFF"

* Actualiza /SwitchableOutput/relay\_1/State en D-Bus.

### F. Nombres de Relé (Opcional)

Tema: rgpio/api/device/MYDEVICE\_001/Relay\_1  
Payload: "Bomba de Agua"

* Actualiza /SwitchableOutput/relay\_1/Name en D-Bus.
* Aparece en la GUI de Venus OS.

## 4. Ejemplo de Flujo Completo

1. Módulo detecta hardware conectado → registra dispositivo en MQTT.
2. Motor Central crea entradas y salidas virtuales en D-Bus.
3. Usuario activa un relé desde GUI → MQTT envía comando al módulo → módulo enciende relé físico.
4. Módulo reporta estado del relé → Motor Central actualiza GUI y D-Bus.
5. Cuando módulo se apaga → envía UNREGISTER → limpieza de entradas/salidas.