

Gosund SP111 Smart Plug

How to reflash with custom firmware using OTA: [TUYA-CONVERT](#) OLD method: [TuyOTA](#)

I was using tasmota-lite.bin v9.2.0 firmware.

```
pushd files
wget tasmota-lite.bin v9.2.0
popd
```

tuya-covert needs small fix for calling iw binary on Debian Buster:

```
=====
Starting smart config pairing procedure
Waiting for the device to install the intermediate firmware
Put device in EZ config mode (blinking fast)
Sending SSID          vtrust-flash
Sending wifiPassword
Sending token          00000000
Sending secret         0101
.....
SmartConfig complete.
Resending SmartConfig Packets
.....
IoT-device is online with ip 10.42.42.42
Fetching firmware backup
  % Total    % Received % Xferd  Average Speed   Time    Time     Time
Current                                  Dload  Upload  Total  Spent  Left
Speed
100 1024k  100 1024k    0     0  69658      0  0:00:15  0:00:15  --:--:--
32846
curl: Saved to filename 'firmware-1a94dd.bin'
=====
Getting Info from IoT-device
VTRUST-FLASH 1.5
(c) VTRUST GMBH https://www.vtrust.de/35c3/
READ FLASH: http://10.42.42.42/backup
ChipID: 1a94dd
MAC: 50:02:91:1A:94:DD
BootVersion: 7
BootMode: normal
FlashMode: 1M DOUT @ 40MHz
FlashChipId: 144051
FlashChipRealSize: 1024K
Active Userspace: user2 0x81000
=====
Ready to flash third party firmware!
```

For your convenience, the following firmware images are already included in this repository:

Tasmota v8.1.0.2 (wifiman)
ESPurna 1.13.5 (base)

You can also provide your own image by placing it in the /files directory
Please ensure the firmware fits the device and includes the bootloader

MAXIMUM SIZE IS 512KB

Available options:

0) return to stock
1) flash espurna.bin
2) flash tasmota.bin
3) flash tasmota-lite.bin
q) quit; do nothing

Please select 0-3: 2

Are you sure you want to flash tasmota.bin? This is the point of no return
[y/N] y

Attempting to flash tasmota.bin, this may take a few seconds...

Flashed http://10.42.42.1/files/tasmota.bin successfully in 11289ms,
rebooting...

Look for a tasmota-xxxx SSID to which you can connect and configure

Be sure to configure your device for proper function!

HAVE FUN!

=====

Do you want to flash another device? [y/N] y

=====

ESPHome Config

Smart Plug - Gosund SP111 Calibrating an ESPHome flashed power plug

[gosund_sp1.yaml](#)

```
esphome:
  name: gosund_pralka
  platform: ESP8266
  board: esp01_1m
  esp8266_restore_from_flash: true

wifi:
  ssid: "myssid"
  password: "mypass"

  # Enable fallback hotspot (captive portal) in case wifi connection
  fails

ap:
```

```
    ssid: "Gniazdko Sp biuro"
    password: "Tmk9tzAUQmr8"

captive_portal:

# Enable logging
logger:

# Enable Home Assistant API
api:

ota:

binary_sensor:
  - platform: gpio
    name: "Power Button"
    pin:
      number: GPIO13
      inverted: true
    on_press:
      - switch.toggle: relay

switch:
  - platform: gpio
    id: relay
    name: ON/OFF
    pin: GPIO15
    on_turn_on:
      - light.turn_on: led
    on_turn_off:
      - light.turn_off: led

time:
  - platform: homeassistant
    id: homeassistant_time
    timezone: Europe/Warsaw
substitutions:
  plug_name: GspBiuro
  # Higher value gives lower watt readout
  current_res: "0.0012"
  # Lower value gives lower voltage readout
  voltage_div: "771"

# Enable Web server
web_server:
  port: 80

sensor:
  - platform: wifi_signal
    name: "${plug_name} - WiFi Signal"
```

```
update_interval: 6s
- platform: uptime
  name: "${plug_name} - Uptime"
  icon: mdi:clock-outline
  update_interval: 60s
- platform: hlw8012
  sel_pin:
    number: GPIO12
    inverted: True
  cf_pin: GPIO05
  cfl_pin: GPIO04
  current_resistor: ${current_res}
  voltage_divider: ${voltage_div}
  current:
    name: "${plug_name} - Ampere"
    unit_of_measurement: A
    accuracy_decimals: 3
    icon: mdi:flash-outline
  voltage:
    name: "${plug_name} - Volt"
    unit_of_measurement: V
    accuracy_decimals: 1
    icon: mdi:flash-outline
  power:
    name: "${plug_name} - Watt"
    unit_of_measurement: W
    id: "${plug_name}_Wattage"
    icon: mdi:flash-outline
  change_mode_every: 4
  update_interval: 3s
- platform: total_daily_energy
  name: "${plug_name} - Consumption"
  power_id: "${plug_name}_Wattage"
  filters:
    # Multiplication factor from W to kW is 0.001
    - multiply: 0.001
  unit_of_measurement: kWh
  icon: mdi:clock-alert

text_sensor:
- platform: version
  name: "${plug_name} - ESPHome Version"

output:
  # Relay state led
- platform: esp8266_pwm
  id: state_led
  pin:
    number: GPIO2
    inverted: true
```

```
light:
  # Relay state light
  - platform: monochromatic
    output: state_led
    id: led

# Uses the red LED as a ESPhome status indicator
status_led:
  pin:
    number: GPIO0
    inverted: true
```

Upgrade from Tasmota:

```
run esphome config.yaml compile
```

Creating BIN file ".pioenvs/gosund_1a5f71/firmware.bin" using
"/root/.platformio/packages/framework-
arduinoespressif8266/bootloaders/eboot/eboot.elf" and
".pioenvs/gosund_1a5f71/firmware.elf"

Or compile and download binary from dashboard.

From:

<https://niziak.spox.org/wiki/> - **niziak.spox.org**

Permanent link:

https://niziak.spox.org/wiki/home_automation:esp8266:hw:gosund_sp111

Last update: **2020/12/20 21:18**

