

MANUALE DI INSTALLAZIONE SONOFF 4 CH ESP8285

PER

DOMOTICZ

SU

RaspberryPi®

Rev0 del 1 ottobre 2017



Sommario

1.	DESCRIZIONE.....	3
1.1	PIN SU SONOFF 4 CH.....	3
2.	RIPROGRAMMAZIONE DEL FIRMWARE	4
2.1	MATERIALE OCCORRENTE	4
2.2	ERASE.....	5
2.3	UPLOAD ESPEASY	6
3.	CONFIGURAZIONE GENERALE SU ESP.....	7
3.1	COLLEGAMENTO ALLA PROPRIA RETE WIFI.....	7
3.2	SCHERMATA MAIN	9
3.3	SCHERMATA CONFIG.....	9
3.4	SCHERMATA HARDWARE	11
3.5	SCHERMATA TOOLS.....	12
3.6	SCHERMATA TOOLS ADVANCED	12
3.7	SCHERMATA DEVICES	13
4.	CONFIGURAZIONE SU DOMOTICZ	14
4.1	CREAZIONE DISPOSITIVO VIRTUALE SONOFF_4_CH	14
4.2	CREAZIONE SENSORI VIRTUALI -> LUCE*	14
4.3	IDENTIFICAZIONE VALORI IDX DEI SENSORI CREATI.....	15
4.4	IMPOSTAZIONE ON ACTION E OFF ACTION SU SENSORI.....	15
4.4.1	LUCE1	16
4.4.2	LUCE2	16
4.4.3	LUCE3	16
4.4.4	LUCE4	17
5.	CONFIGURAZIONE SU ESP	17
5.1.1	DEVICE LUCE1	17
5.1.2	DEVICE LUCE2	18
5.1.3	DEVICE LUCE3	19
5.1.4	DEVICE LUCE4.....	19
5.3.1	LED WIFI	22

1. DESCRIZIONE

Il SONOFF 4CH è un dispositivo a guida DIN che permette di controllare via WIFI 4 carichi in modo da accenderli e spegnerli. La commutazione è realizzata tramite 4 relè. A bordo del dispositivo ci sono anche 4 pulsanti che permettono di accendere e spegnere i relativi relè. Ad ognuno dei relè è collegato un led rosso che si accende di conseguenza all'accensione del relè.



Il dispositivo può essere riprogrammato con ESPEASY prestando attenzione che il chip in questo caso non è il solito ESP8266, ma ESP8285. Pertanto bisogna disporre di un file bin del FW adeguato o utilizzare Arduino IDE settando il device corretto. Nel mio caso ho preferito usare il file già previsto per ESP8285.

1.1 PIN SU SONOFF 4 CH

GPIO-0	Pulsante Canale 1 e per upload FW
GPIO-12	Relay + LED Channel 1, Active High IO12
GPIO-9	Pulsante Channel 2 IO9
GPIO-5	Relay + LED Channel 2, Active High IO5
GPIO-10	Pulsante Channel 3 IO10
GPIO-4	Relay + LED Channel 3, Active High IO4
GPIO-14	Pulsante Channel 4 IO14
GPIO-15	Relay + LED Channel 4, Active High IO15
GPIO-7	Non collegato, spazio vuoto per una presa da 2,5 mm (vedi TH16) GPIO07
GPIO-8	Non collegato, spazio vuoto per una presa da 2,5 mm (vedi TH16) GPIO08
GPIO-13	Led WIFI blu Active Low

2. RIPROGRAMMAZIONE DEL FIRMWARE

2.1 MATERIALE OCCORRENTE

Per riprogrammare il SONOFF occorre:

1. SONOFF 4CH
2. Pc portatile o fisso con scheda WIFI
3. Saldatore di bassa potenza
4. Stagno e pin strip da 5 pin da saldare sulla basetta del SONOFF 4 CH
5. Programmatore FT232BL (attenzione se CLONE non funziona)
6. Drivers per FT232BL (scaricabili da qui <http://www.ftdichip.com/Drivers/VCP.htm>)
7. ESP8266Flasher (scaricabile da qui <https://github.com/nodemcu/nodemcu-flasher>)
8. Flash vuota per pulire l'ESP8266 del SONOFF (qui http://www.pratikpanda.com/wp-content/uploads/2016/05/blank_1MB.zip)
9. Firmware ESPEASY 4 CH (https://github.com/blddk/esp8285/tree/sonoff_4ch)

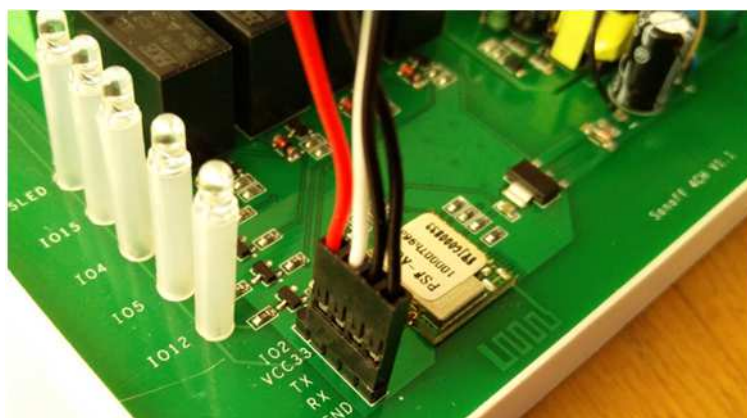
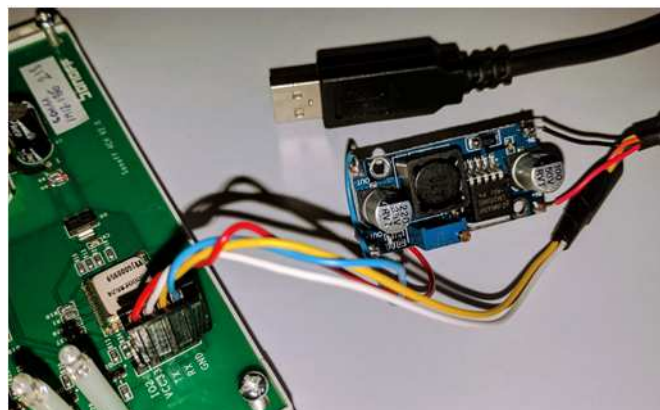
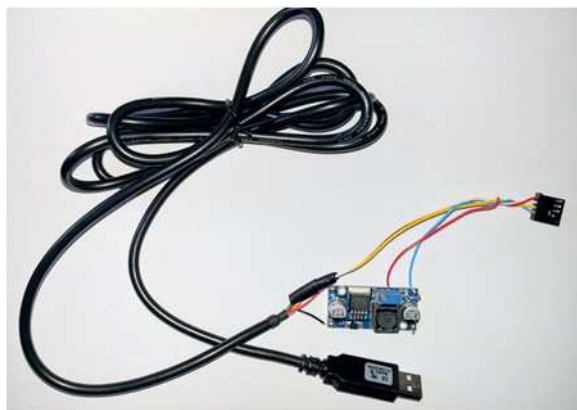
Scaricare i software e firmware e salvarli ognuno in una cartella sul desk o dove fa comodo

Procediamo a saldare i 5 pin sulla schedina del SONOFF

Installare i drivers (punto 6) sul PC e collegare l'FT232BL (**Attenzione che il positivo sia a 5V**)

Controllare in gestione dispositivi la porta COM su cui si è installato (nel mio caso su COM11).

PS Per portare la tensione da 5V del filo rosso che esce dal programmatore FTDI a 3,3V ho inserito un riduttore di tensione.



Piedini SONOFF
GPIO14
MASSA
TX
RX
+ 3,3 V

2.2 ERASE

Collegare cavetto programmatore FTDI alla porta USB.

Pigiare e tenendo pigiato il pulsante 1 sul SONOFF 4 CH collegare il cavetto al SONOFF 4 CH, attendere qualche secondo e quindi rilasciare il pulsante.

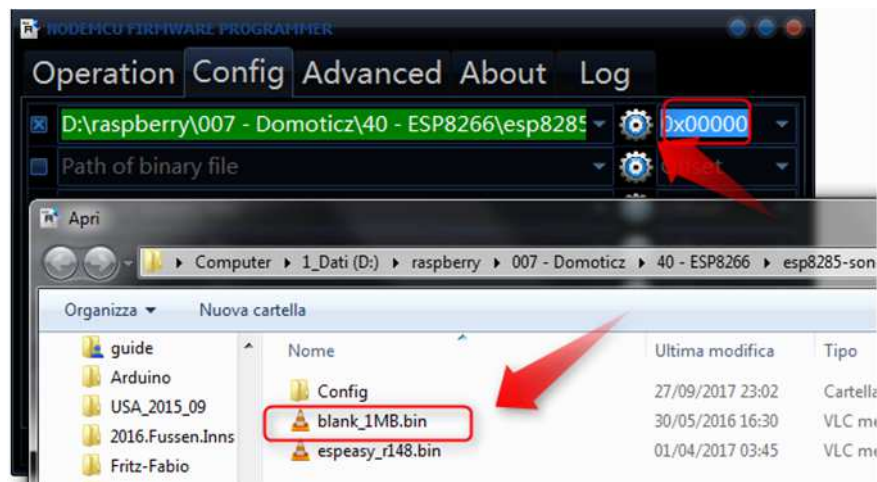
Entrare nella cartella **esp8285-sonoff_4ch**

Lanciare **ESP8266Flasher.exe** (utente normale no amministratore)

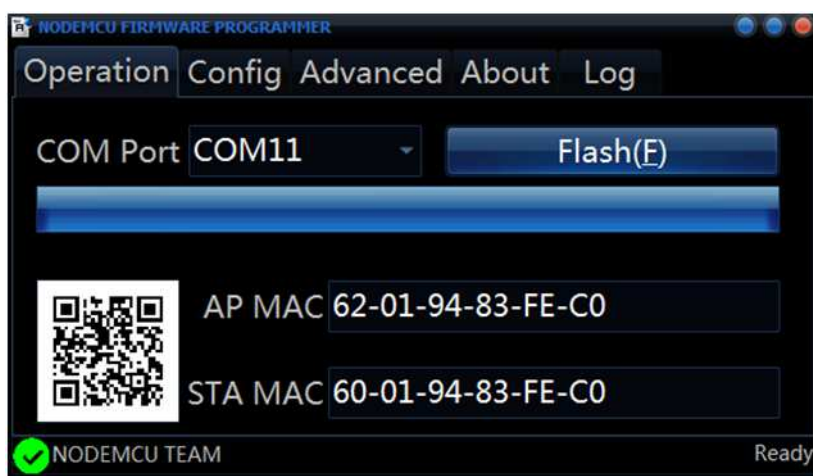
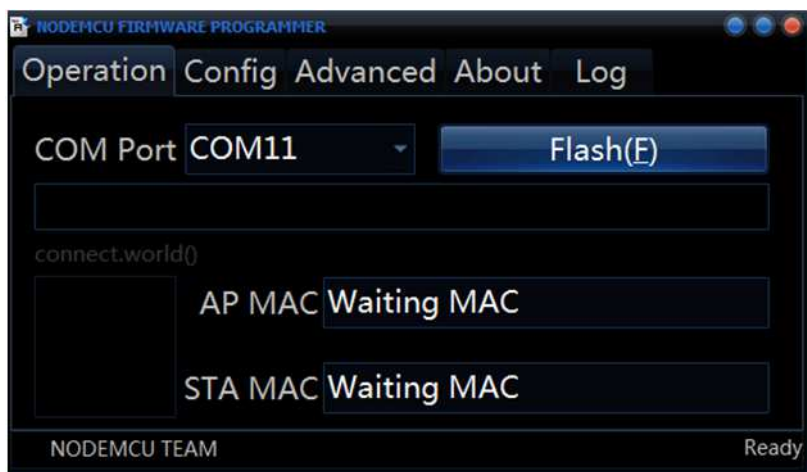
In Advanced settare



In Config selezionare il file



In Operation la porta dovrebbe selezionarsi in automatico nel mio caso COM11



Premere Flash e attendere il completamento dell'operazione. In basso a SX simbolo verde. Chiudere il SW e disalimentare SONOFF.

2.3 UPLOAD ESPEASY

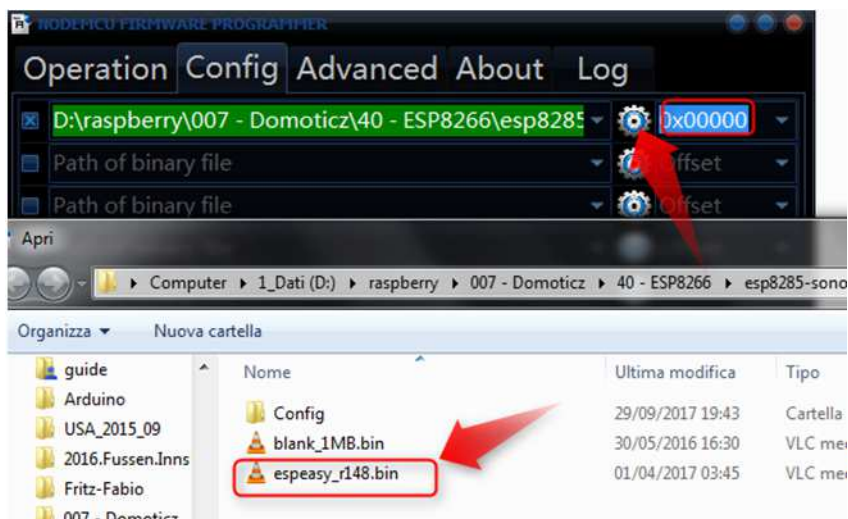
Ripartire da zero. Staccare cavetto programmatore FTDI alla porta USB.

Ricollegare cavetto programmatore FTDI alla porta USB.

Pigiare e tenendo pigiato il pulsante 1 sul SONOFF 4 CH, collegare il cavetto al SONOFF 4 CH, attendere qualche secondo e quindi rilasciare il pulsante.

Lanciare **ESP8266Flasher.exe**

Con gli stessi settaggi di prima andare in Config e selezionare il file del FW

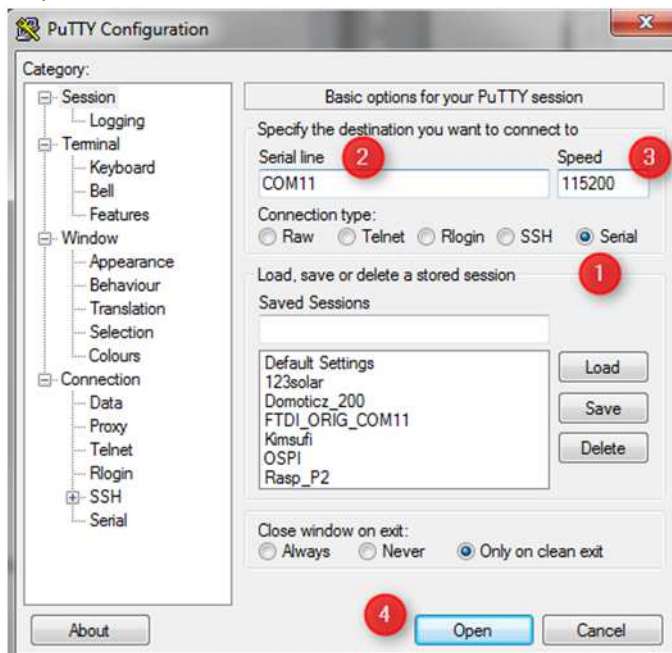


Premere Flash e attendere il completamento dell'operazione. In basso a SX simbolo verde. Chiudere il SW e disalimentare SONOFF 4 CH.

3. CONFIGURAZIONE GENERALE SU ESP

3.1 COLLEGAMENTO ALLA PROPRIA RETE WIFI

Dopo aver sconnesso SONOFF, con FTDI connessa a USB, avviare Putty



Settare come da figura e premere Open (4)

Ricollegare alla rete il SONOFF al programmatore FTDI

Su Putty si vedranno scorrere delle righe fino che si fermerà con una schermata simile


```
COM11 - PuTTY
FLASH: Zero Fill Sector: 238
FLASH: Zero Fill Sector: 239
FLASH: Zero Fill Sector: 240
FLASH: Zero Fill Sector: 241
FLASH: Zero Fill Sector: 242
FLASH: Zero Fill Sector: 243
FLASH: Zero Fill Sector: 244
FLASH: Zero Fill Sector: 245
FLASH: Zero Fill Sector: 246
FLASH: Zero Fill Sector: 247
FLASH: Zero Fill Sector: 248
FLASH: Zero Fill Sector: 249
FLASH: Zero Fill Sector: 250
FLASH: Settings saved
FLASH: Settings saved

ets Jan  8 2013,rst cause:1, boot mode:(3,7)

load 0x4010f000, len 1384, room 16
tail 8
chksum 0x2d
csum 0x2d
v09826c6d
~ld
^U
  INIT : Booting Build nr:120
WIFI : No SSID!
INIT : I2C
INIT : Boot OK
INIT : Reboot from deepsleep
WD   : Uptime 0 ConnectFailures 0 FreeMem 27016
```

Aspettare un minuto e controllare nelle reti WIFI, dovrebbe comparire una nuova rete con nome ESP_0
Cliccarci sopra per collegarsi ad essa (password **configesp**)

Lanciare il browser e inserire l'indirizzo 192.168.4.1

Nella pagina che compare flaggare la propria WIFI ed inserire la password.
Poi cliccare su Connect

Welcome to ESP Easy: newdevice

Wifi Setup wizard

- ☐ tyson
- ☒ FEB_01
- ☐ FEB_L2.4GHz
- ☐ other SSID:

Password:

inserire PSW

il SONOFF 4 CH si riavvierà

Welcome to ESP Easy: newdevice

Wifi Setup wizard

Please wait for

16..

seconds while trying to connect

e alla fine del conto alla rovescia (in alto compare l'indirizzo IP), connettersi alla propria rete WIFI

Welcome to ESP Easy: newdevice

ESP is connected and using IP Address: 192.168.0.247

Connect your laptop / tablet / phone back to your main Wifi network and [Proceed to main config](#)

Powered by www.esp8266.nu

Accedere all'indirizzo IP nel mio caso 192.168.0.186.

3.2 SCHERMATA MAIN

In alto in azzurro ci sono i vari menu che andremo a vedere successivamente uno x uno. Sotto i menu ci sono tutti i dati riepilogativi della parte rete, versione firm, numero periferica, ecc

Welcome to ESP Easy: SonOff

[Main](#) [Config](#) [Hardware](#) [Devices](#) [Tools](#)

System Info		
Load:	2% (LC=34237)	
Uptime:	2 minutes	
Wifi RSSI:	-65 dB	
IP:	192.168.0.230	
GW:	192.168.0.41	
Build:	120	
Core Version:	2_3_0	
Unit:	0	
STA MAC:	5c:cf:7f:af:4f:1f	
AP MAC:	5e:cf:7f:af:4f:1f	
ESP Chip ID:	11489055	
Flash Chip ID:	1327328	
Flash Size:	1024 kB	
Sketch Size/Free:	401 kB / 536 kB	
Free Mem:	28512	
Boot cause:	Cold boot	
Node List:	IP	Age

3.3 SCHERMATA CONFIG

In questa schermata si configura ESP con DOMOTICZ e si setta un indirizzo IP fisso e Unit nr. 18.

IP: **192.168.0.218**

Name: **ESP_SONOFF_4CH**

Welcome to ESP Easy: SonOff

[Main](#) [Config](#) [Hardware](#) [Devices](#) [Rules](#) [Tools](#)

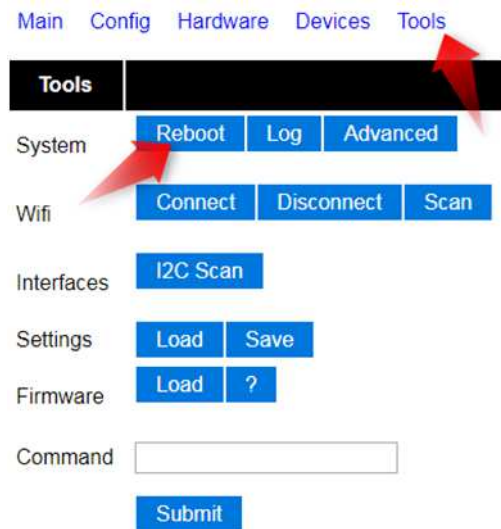
Main Settings	
Name:	<input type="text" value="SonOff"/>
Admin Password:	<input type="text" value="philips63"/>
SSID:	<input type="text" value="FEB_L2.4GHz"/>
WPA Key:	<input type="text"/>
WPA AP Mode Key:	<input type="text" value="configesp"/>
Unit nr:	<input type="text" value="13"/>
Protocol:	<input type="text" value="Domoticz HTTP"/> ?
Locate Controller:	<input type="text" value="Use IP address"/>
Controller IP:	<input type="text" value="192.168.0.200"/>
Controller Port:	<input type="text" value="8080"/>
Controller User:	<input type="text" value="febpro"/>
Controller Password:	<input type="text"/>
Sensor Delay:	<input type="text" value="60"/>
Sleep Mode:	<input type="checkbox"/> ?

Optional Settings	
ESP IP:	<input type="text" value="192.168.0.213"/>
ESP GW:	<input type="text" value="192.168.0.1"/>
ESP Subnet:	<input type="text" value="255.255.255.0"/>
ESP DNS:	<input type="text" value="192.168.0.1"/>

- Name: impostare nome periferica per un più facile riconoscimento
- Admin: Password impostazione password accesso periferica
- SSID: WIFI network SSID – dovrebbe essere già impostata
- WPA Key: WPA WIFI Access Point – dovrebbe essere già impostata
- WPA AP Mode Key: è la password WPA di accesso ad ESPEasy
- Unit nr: inserire un numero univoco della periferica (range 1-31) importante
inserire un valore ed evitare doppioni
- Protocol: selezionare il protocollo di comunicazione con il programma Domoticz nel raspberry.
- Locate Controller: impostare il tipo di comunicazione con il protocollo.
- Controller IP: Selezionare l'IP del raspberry dove installato il programma domotico
- Controller Port: Selezionare la porta del raspberry dove installato il programma domotico.
- Controll User: inserire login di Domoticz
- Controll Password: inserire la password di accesso a Domoticz
- Sensor Delay: Imposta il ritardo tra segnalazione sensore secondi.
- Sleep Mode: selezionare il flag per settare l'ESP a bassa potenza (sembrerebbe bloccare

- ESP IP: l'ESP, quindi per il momento non selezionarlo) per info SleepMode IP fisso da assegnare
- ESP GW: IP router
- ESP Subnet: 255.255.255.0
- ESP DNS: DNS server (IP router)

Al termine selezionare SUBMIT per confermare. Poi andare su Tools e quindi premere Reboot



A questo punto il SONOFF si riavvierà cambiando IP impostandosi su quello inserito.

Nel mio caso:

192.168.0.218

A questo punto cambiare IP sul browser con quello inserito in Config ESP IP

Su PuTTY si può vedere il cambio di IP

```
INIT : Booting Build nr:120
IP   : Static IP :192.168.0.214
WIFI : Connecting... 1
WIFI : Connected!
INIT : I2C
INIT : Boot OK
INIT : Normal boot
EVENT: System#Boot
```

3.4 SCHERMATA HARDWARE

La schermata HW permette di configurare alcuni PIN assegnati di default ad esempio TX, RX, ecc. Infatti è possibile di "liberarli" per essere usati per collegare altri sensori.

La perdita dei TX ed RX non è importante in quanto ESPEasy è possibile riprogrammarlo tramite WIFI in Ota. Quindi, SDA selezionare il pin per I2C (andremo a mettere 1 ex TX vedi schermata tools), SCL selezionare il clock per I2C (andremo a mettere 3 ex RX vedi schermata tools) Pin mode x Selezionare uno stato di boot opzionale per questo GPIO.

I GPIO vengono usati per collegare i sensori di ESPEasy. È possibile collegare più sensori per gli stessi due fili.

Non dimenticare di usare resistori pull-up sia su SDA e SCL, a meno che uno dei sensori non li abbia già montati a bordo.

[Main](#) [Config](#) [Hardware](#) [Devices](#) [Tools](#)

Hardware Settings

Wifi Status Led:

SDA:

SCL:

GPIO boot states:

Pin mode 0:

Pin mode 2:

Pin mode 4:

Pin mode 5:

Pin mode 9:

Pin mode 10:

Pin mode 12:

Pin mode 13:

Pin mode 14:

Pin mode 15:

Pin mode 16:

3.5 SCHERMATA TOOLS

System ed WIFI sono chiari, la cosa importante è Advanced che viene spiegata al prossimo Riquadro.

Interfaces fa lo scan dei sensori I2C (comodo per vedere se impostati bene in Hardware).

Setting load e save configurazione

Firmware per caricare in OTA il firmware precedentemente scaricato sul pc

Submit per confermare

3.6 SCHERMATA TOOLS ADVANCED

Ho semplicemente

- 1) Deflaggato Enable serial port per utilizzare i GPIO1 e 3 al posto di TX ed RX
- 2) Impostato la porta UDP a 65500
- 3) Flaggato Global Sync per connettere tutti gli ESP in rete

Welcome to ESP Easy: SonOff

[Main](#) [Config](#) [Hardware](#) [Devices](#) [Tools](#)

Advanced Settings	
Subscribe Template:	<input type="text"/>
Publish Template:	<input type="text"/>
Message Delay (ms):	<input type="text" value="1000"/>
Fixed IP Octet:	<input type="text" value="0"/>
Use NTP:	<input type="checkbox"/>
NTP Hostname:	<input type="text"/>
Timezone Offset: (Minutes)	<input type="text" value="0"/>
DST:	<input type="checkbox"/>
Syslog IP:	<input type="text" value="0.0.0.0"/>
Syslog Level:	<input type="text" value="0"/>
UDP port:	<input type="text" value="65500"/> 2
Enable Serial port:	<input type="checkbox"/> 1
Serial log Level:	<input type="text" value="2"/>
Web log Level:	<input type="text" value="2"/>
Baud Rate:	<input type="text" value="115200"/>
WD I2C Address:	<input type="text" value="0"/>
Custom CSS:	<input type="checkbox"/>
Use SSDP:	<input type="checkbox"/>
Connection Failure Threshold:	<input type="text" value="0"/>
Rules:	<input type="checkbox"/>
Experimental Settings	
I2C ClockStretchLimit:	<input type="text" value="0"/>
Global Sync:	<input checked="" type="checkbox"/> 3
<input type="button" value="Submit"/>	

3.7 SCHERMATA DEVICES

Questa è la parte più importante. Con le frecce DX e SX è possibile spostarsi in più pagine, in tutto 12 devices.

[Main](#) [Config](#) [Hardware](#) [Devices](#) [Tools](#)

<	>	Task	Device	Name	Port	IDX/Variable	GPIO	Values
Edit		5						
Edit		6						
Edit		7						
Edit		8						

Cliccando su Edit si entra nella configurazione del Devices.

Al primo ingresso apparirà la semplice schermata per selezionare con menù a tendina il devices voluto.

I menu comuni dei vari devices sono

- Device: selezionare con il menù a tendina il sensore voluto [qui](#) l'elenco
- Nome: il nome del sensore
- Delay: io metto sempre a 60 tranne i pulsanti che metto a 0
- IDX/Var: è il numero univoco del device in Domoticz (mettere un numero a caso e successivamente vedere l'idx che assegna Domoticz e cambiarlo con lo stesso idx)
- Global sinc: serve per sincronizzare tutti gli ESPEasy nella rete di casa
- Formula e Decimal: si assegna una formula per modificare il valore o impostare i decimali Value
- Name: il nome che riporta nella schermate dei Devices

Altri campi si possono valorizzare/apparire in funzione del sensore impostato.

Cliccare su SUBMIT per confermare CLOSE per uscire senza confermare.

4. CONFIGURAZIONE SU DOMOTICZ

4.1 CREAZIONE DISPOSITIVO VIRTUALE SONOFF_4_CH

Su menù Hardware di Domoticz aggiungere dispositivo DUMMY

The screenshot shows the 'Hardware' configuration page in Domoticz. The 'Enabled' checkbox is checked. The 'Name' field contains 'Sonoff_4_CH'. The 'Type' dropdown is set to 'Dummy (Does nothing, use for virtual switches only)'. The 'Data Timeout' dropdown is set to 'Disabled'. Below the dropdowns, there is a warning: 'Specifying a Data Timeout will restart the hardware device if no data is time. Do not enable this option for devices that do not receive data!'. At the bottom, there is a blue 'Add' button.

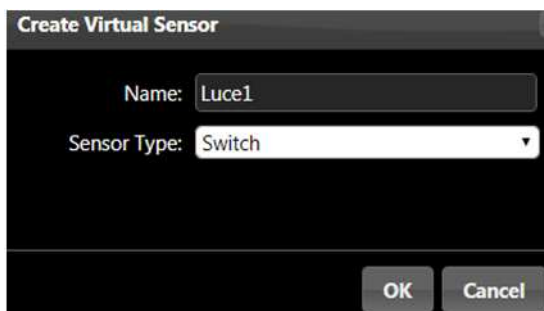
4.2 CREAZIONE SENSORI VIRTUALI -> LUCE*

Premi Create Virtual Sensor

The screenshot shows a table of virtual sensors. The first row has ID 13, Name 'Sonoff_4_CH', and a 'Yes' status. The second row has ID 12, Name 'PVOutput', and a 'Yes' status. To the right of the table, there is a text box that says 'Dummy (Does nothing, use for virtual switches only)' and a blue button labeled 'Create Virtual Sensors'. A large red arrow points from the 'Yes' status of the second row to the 'Create Virtual Sensors' button.

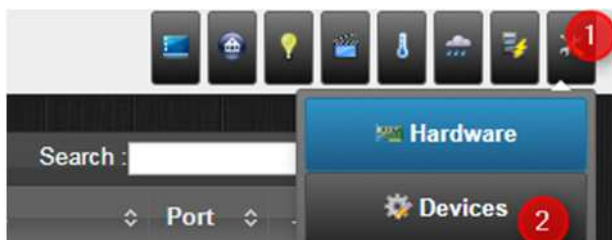
13	Sonoff_4_CH	Yes
12	PVOutput	Yes

E crea 4 Sensori Virtuali , Luce1, Luce2, Luce3 e Luce4 selezionando su Sensor Type Switch



4.3 IDENTIFICAZIONE VALORI IDX DEI SENSORI CREATI

Vai su Devices



e cerca i valori di Idx relativi ai 4 Sensori creati

Idx	Hardware	ID	Unit	Name	Type	SubType	Data	Last Seen
80	Sonoff_4_CH	000140A0	1	Luce1	Light/Switch	Switch	Off	2017-09-30 09:17:10
83	Sonoff_4_CH	000140A3	1	Luce4	Light/Switch	Switch	Off	2017-09-30 09:06:07
82	Sonoff_4_CH	000140A2	1	Luce3	Light/Switch	Switch	Off	2017-09-30 08:50:17
81	Sonoff_4_CH	000140A1	1	Luce2	Light/Switch	Switch	Off	2017-09-30 08:47:50

Luce1 Idx 80

Luce2 Idx 81

Luce3 Idx 82

Luce4 Idx 83

4.4 IMPOSTAZIONE ON ACTION E OFF ACTION SU SENSORI

Ad ogni LuceX su Domoticz,

inserire su On Action


<http://192.168.0.218/control?cmd=GPIO,12,1>

e SU inserire su Off Action

<http://192.168.0.218/control?cmd=GPIO,12,0>

dove GPIO,12,1 varia a seconda della Luce


4.4.1 LUCE1

Name:	<input type="text" value="Luce1"/>
Switch Type:	<input type="text" value="On/Off"/>
Switch Icon:	<div><div>Light/Switch A Lamp or Switch</div></div>
On Delay:	<input type="text" value="0"/> (Seconds) 0 = Disabled
Off Delay:	<input type="text" value="0"/> (Seconds) 0 = Disabled
On Action:	<input type="text" value="http://192.168.0.218/control?cmd=GPIO,12,1"/>
Off Action:	<input type="text" value="http://192.168.0.218/control?cmd=GPIO,12,0"/>
Protected:	<input type="checkbox"/>

<http://192.168.0.218/control?cmd=GPIO,12,1>

<http://192.168.0.218/control?cmd=GPIO,12,0>


4.4.2 LUCE2

Name:	<input type="text" value="Luce2"/>
Switch Type:	<input type="text" value="On/Off"/>
Switch Icon:	<div><div>Light/Switch A Lamp or Switch</div></div>
On Delay:	<input type="text" value="0"/> (Seconds) 0 = Disabled
Off Delay:	<input type="text" value="0"/> (Seconds) 0 = Disabled
On Action:	<input type="text" value="http://192.168.0.218/control?cmd=GPIO,5,1"/>
Off Action:	<input type="text" value="http://192.168.0.218/control?cmd=GPIO,5,0"/>
Protected:	<input type="checkbox"/>

<http://192.168.0.218/control?cmd=GPIO,5,1>

<http://192.168.0.218/control?cmd=GPIO,5,0>

4.4.3 LUCE3

Name:	<input type="text" value="Luce3"/>
Switch Type:	<input type="text" value="On/Off"/>
Switch Icon:	<div><div>Light/Switch A Lamp or Switch</div></div>
On Delay:	<input type="text" value="0"/> (Seconds) 0 = Disabled
Off Delay:	<input type="text" value="0"/> (Seconds) 0 = Disabled
On Action:	<input type="text" value="http://192.168.0.218/control?cmd=GPIO,4,1"/>
Off Action:	<input type="text" value="http://192.168.0.218/control?cmd=GPIO,4,0"/>
Protected:	<input type="checkbox"/>


<http://192.168.0.218/control?cmd=GPIO,4,1>

<http://192.168.0.218/control?cmd=GPIO,4,0>

4.4.4 LUCE4

Name:

Switch Type:

Switch Icon:  **Light/Switch**
A Lamp or Switch

On Delay: (Seconds) 0 = Disabled

Off Delay: (Seconds) 0 = Disabled

On Action:

Off Action:

Protected: ☐

<http://192.168.0.218/control?cmd=GPIO,15,1>

<http://192.168.0.218/control?cmd=GPIO,15,0>

5. CONFIGURAZIONE SU ESP

5.1 CREAZIONE DEVICE SU ESP FUNZIONAMENTO A PULSANTE

Seleziona Devices

Main Config Hardware **Devices** Rules Tools

<	>	Task	Device	Name	Port	IDX/Variable	GPIO	Values
Edit		1						
Edit		2						
Edit		3						
Edit		4						

e crea su ESP i Devices

5.1.1 DEVICE LUCE1

GPIO-0 Pulsante Canale 1 e per upload FW

GPIO-12 Relay + LED Channel 1, Active High IO12

Task Settings	Value
Device:	<input type="text" value="Switch input"/>
Name:	<input type="text" value="Luce1"/>
Delay:	<input type="text" value="0"/> (Op
IDX / Var:	<input type="text" value="80"/>
1st GPIO:	<input type="text" value="GPIO-0 (D3)"/> ▼
Pull UP:	<input checked="" type="checkbox"/>
Inversed:	<input type="checkbox"/>
Switch Type:	<input type="text" value="Switch"/> ▼
Switch Button Type:	<input type="text" value="Push Button Active Low"/> ▼
Send Boot state:	<input checked="" type="checkbox"/>
Send Data:	<input checked="" type="checkbox"/>
Global Sync:	<input type="checkbox"/>

Optional Settings	Value
Value Name 1:	<input type="text" value="L1"/>

5.1.2 DEVICE LUCE2

GPIO-9 Pulsante Channel 2 IO9

GPIO-5 Relay + LED Channel 2, Active High IO5

Task Settings	Value
Device:	<input type="text" value="Switch input"/>
Name:	<input type="text" value="Luce2"/>
Delay:	<input type="text" value="0"/> (Op
IDX / Var:	<input type="text" value="81"/>
1st GPIO:	<input type="text" value="GPIO-9 (D11)"/> ▼
Pull UP:	<input checked="" type="checkbox"/>
Inversed:	<input type="checkbox"/>
Switch Type:	<input type="text" value="Switch"/> ▼
Switch Button Type:	<input type="text" value="Push Button Active Low"/> ▼
Send Boot state:	<input checked="" type="checkbox"/>
Send Data:	<input checked="" type="checkbox"/>
Global Sync:	<input type="checkbox"/>

Optional Settings	Value
Value Name 1:	<input type="text" value="L2"/>

5.1.3 DEVICE LUCE3

GPIO-10 Pulsante Channel 3 IO10

GPIO-4 Relay + LED Channel 3, Active High IO4

Task Settings	Value
Device:	Switch input
Name:	Luce3
Delay:	0 (Op
IDX / Var:	82
1st GPIO:	GPIO-10 (D12) ▾
Pull UP:	<input checked="" type="checkbox"/>
Inversed:	<input type="checkbox"/>
Switch Type:	Switch ▾
Switch Button Type:	Push Button Active Low ▾
Send Boot state:	<input checked="" type="checkbox"/>
Send Data:	<input checked="" type="checkbox"/>
Global Sync:	<input type="checkbox"/>

Optional Settings	Value
Value Name 1:	L3

5.1.4 DEVICE LUCE4

GPIO-14 Pulsante Channel 4 IO14

GPIO-15 Relay + LED Channel 4, Active High IO15

Task Settings	Value
Device:	<input type="text" value="Switch input"/>
Name:	<input type="text" value="Luce4"/>
Delay:	<input type="text" value="0"/> (Opt)
IDX / Var:	<input type="text" value="83"/>
1st GPIO:	<input type="text" value="GPIO-14 (D5)"/>
Pull UP:	<input checked="" type="checkbox"/>
Inversed:	<input type="checkbox"/>
Switch Type:	<input type="text" value="Switch"/>
Switch Button Type:	<input type="text" value="Push Button Active Low"/>
Send Boot state:	<input type="checkbox"/>
Send Data:	<input checked="" type="checkbox"/>
Global Sync:	<input checked="" type="checkbox"/>

Optional Settings	Value
Value Name 1:	<input type="text" value="L4"/>

A questo punto su schermata Devices si vedrà

[Main](#) [Config](#) [Hardware](#) [Devices](#) [Rules](#) [Tools](#)

<	>	Task	Device	Name	Port	IDX/Variable	GPIO	Values
Edit		1	Switch input	Luce1		80	GPIO-0	L1: 0
Edit		2	Switch input	Luce2		81	GPIO-9	L2: 0
Edit		3	Switch input	Luce3		82	GPIO-10	L3: 0
Edit		4	Switch input	Luce4		83	GPIO-14	L4: 0

5.2 ASSEGNAZIONE REGOLE

Se non attivate precedentemente, su Tools, Advances, abilitare Rules

Poi, inserire le seguenti regole in Rules

```
// RELE 1
on Luce1#L1 do
  if [Luce1#L1]=1
    gpio 12,1 // Rele 1 ON
  else
    gpio 12,0 // Rele 1 OFF
  endif
endon

// RELE 2
on Luce2#L2 do
  if [Luce2#L2]=1
    gpio 5,1 // Rele 1 ON
  else
    gpio 5,0 // Rele 1 OFF
  endif
endon

// RELE 3
on Luce3#L3 do
  if [Luce3#L3]=1
    gpio 4,1 // Rele 1 ON
  else
    gpio 4,0 // Rele 1 OFF
  endif
endon

// RELE 4
on Luce4#L4 do
  if [Luce4#L4]=1
    gpio 15,1 // Rele 1 ON
  else
    gpio 15,0 // Rele 1 OFF
  endif
endon
```

Purtroppo ESP non ha un controllo di sintassi e se si sbaglia qualcosa la regola non funziona e basta.

Per verificare se la regola Luce*#L* funziona bisogna eliminare temporaneamente su Luce* il suo IDX di Domoticz, altrimenti il comando del pulsante viene inviato a Domoticz che poi attiva l'evento il GPIO di ESP. In questo caso se manca il collegamento WIFI con Domoticz con il pulsante non possiamo accendere o spegnere il rele. Dopo il test della regola rimettere IDX su Luce.

5.3 FUNZIONAMENTO A INTERRUTTORE O DEVIATORE

Finora abbiamo programmato il nostro dispositivo con funzionamento a pulsante.

Questo significa che tramite una pressione momentanea sui pulsanti presenti sul circuito del SONOFF 4CH si accende il relativo relè e alla seconda pressione si spegne.

Collegando in parallelo a questi pulsanti dei pulsanti esterni del nostro impianto elettrico è possibile sincronizzare Domoticz e il nostro impianto di casa. In pratica all'accensione di una lampada tramite pulsante sul muro di casa si vedrà la sua attivazione su Domoticz.

Se al contrario, nel nostro impianto di casa non ci sono pulsanti, ma interruttori o catene di deviatori (invertitori, ecc) è sufficiente cambiare qualche configurazione su ESP SONOFF 4CH. La configurazione è per ogni singola luce e quindi si possono avere in contemporanea configurazioni diverse su luci diverse. Es. Luce1 a luce Luce2 a interruttore (deviatore).

In particolare andare su Devices, selezionare il device Luce* e su Switch Button Type selezionare Normal Switch

Task Settings	Value
Device:	Switch input
Name:	Luce1
Delay:	0 (Op)
IDX / Var:	80
1st GPIO:	GPIO-0 (D3)
Pull UP:	<input checked="" type="checkbox"/>
Inversed:	<input type="checkbox"/>
Switch Type:	Switch
Switch Button Type:	Normal Switch
Send Boot state:	<input checked="" type="checkbox"/>
Send Data:	<input checked="" type="checkbox"/>
Global Sync:	<input type="checkbox"/>

Optional Settings	Value
Value Name 1:	L1

In questo caso i pulsanti a bordo del SONOFF 4 CH non essendo interruttori non funzionano regolarmente, ma devono essere mantenuti premuti o rilasciati.

Ora si può collegare la catena dei deviatori (invertitori ecc..) in parallelo ai pulsanti del SONOFF 4 CH in modo che questi chiudano il circuito.

5.3.1 LED WIFI

Il led del WIFI è collegato al GPIO13. Il led BLU si accende se inviato il comando LOW