

# Tasmota - Anpassung durch user\_config

## Inhaltsverzeichnis

- [I - Schnellstart mit user\\_config\\_override.h](#)
- [II - Erklärung und Hinweise](#)
  - [II.I - my\\_user\\_config.h](#)
  - [II.II - user\\_config\\_override.h](#)
- [III - FAQ](#)
  - [III.I - Welche Einstellungen sind vordefiniert?](#)
  - [III.II - Kann ich meine bearbeitete user\\_config\\_override.h wiederverwenden?](#)
  - [III.III - Warum immer paarweise #undef...#define ?](#)
  - [III.IV - Komplexere Bedingungen bildet man mit:](#)

In diesem Abschnitt wird die **Anpassung von Tasmota** mit eigenen Einstellungen durch die Dateien [my\\_user\\_config.h](#) und [user\\_config\\_override.h](#) beschrieben.

## I - Schnellstart mit user\_config\_override.h

**Voraussetzung:**

- **Atom** / **VSC** Installation mit PlatformIO oder **Arduino IDE**
- **Tasmota** Source-Code z.B. [Sonoff-Tasmota-development](#)

[MINIMAL\\_user\\_config\\_override.h.zip](#)

- für das schnelle Loslegen - wenige Einträge für eine Basis-Konfiguration

[UNIVERSAL\\_user\\_config\\_override.h.zip](#)

- alle wichtigen Zeilen für die eigene Konfiguration - wie in der [my\\_user\\_config.h](#)

1. [user\\_config\\_override.h](#) in den Ordner `\sonoff` kopieren
2. nur in [user\\_config\\_override.h](#) die Einstellungen auf **eigene Konfiguration** ändern
3. [user\\_config\\_override.h](#) **aktivieren** (Details weiter unten) und [Tasmota](#) kompilieren

## II - Erklärung und Hinweise

### II.I - my\_user\_config.h

In der Datei `sonoff\my_user_config.h` werden die Einstellungen und Parameter für die Funktion von [Tasmota](#) festgehalten.

Die [my\\_user\\_config.h](#) (bis [Tasmota](#) v6.3.0 `user_config.h`) wird in zwei Abschnitte unterteilt:

- **Abschnitt 1 (SECTION 1):**

Darin sind Konfigurationsdaten für **WLAN**, **MQTT**, **Zeitzone**, **GeoPosition** usw. enthalten.

Alle hier vorgenommenen Einstellungen werden bei einem Update normalerweise **nicht** überschrieben und bleiben im Flash-Speicher **erhalten**.

Erst eine Änderung des CFG HOLDER oder das Löschen des Flash-Speichers bewirkt ein Überschreiben.

Mit Befehlen über die Konsole (Web / Seriell) oder MQTT können die Parameter **permanent** geändert bzw. **gespeichert** werden.

- **Abschnitt 2 (SECTION 2):**

Hier können **Features und Sensoren** aktiviert und deaktiviert werden, um Speicherplatz zu sparen.

Bei jedem Update, mit Änderungen in diesem Abschnitt, wird die **vorherige Konfiguration überschrieben**.

## II.II - user\_config\_override.h

Es ist zwar möglich alle gewünschten persönlichen Einstellungen in der [my\\_user\\_config.h](#) vorzunehmen,

**ALLE eigenen Parameter und Anpassungen** sollte man aber in der **sonoff**[user\\_config\\_override.h](#) eintragen.

### Warum sollte man das tun?

[my\\_user\\_config.h](#) wird i. d. R. bei jedem [Tasmota](#)-Update **verändert**.

- Anpassungen müssen dann **jedes Mal neu** in der [my\\_user\\_config.h](#) vorgenommen und abgeglichen werden.

+ [user\\_config\\_override.h](#) kann bedenkenlos **von der einen zur anderen** [Tasmota](#)-Version mitgenommen werden

+ dabei sind, je nach Bedarf, keinerlei Änderungen notwendig

+ sie kann ganz auf persönliche Anforderungen zugeschnitten und übersichtlich gehalten werden

+ was man nie braucht, wird ausgeklammert

+ gleichbleibende Einstellungen können unverändert belassen werden

+ für Spezialfälle kann man verschiedene [user\\_config\\_override.h](#) Dateien anlegen

### Wie funktioniert das?

Die [my\\_user\\_config.h](#) wird beim Kompilieren in Atom / VSC / Arduino-IDE **zuerst** abgearbeitet.

Erst **danach** werden die Definitionen in der [user\\_config\\_override.h](#) berücksichtigt.

Das bedeutet, was zuvor in der [my\\_user\\_config.h](#) mit `#define` definiert wurde,

kann in der [user\\_config\\_override.h](#)

--> **deaktiviert** werden mit `#undef` oder

--> **neu definiert** werden mit `#undef / #define`

### Wie macht man das?

Bei jeder neuen Version von [Tasmota](#):

1. [user\\_config\\_override.h](#) in den Ordner `\sonoff` kopieren

Dazu wählt man z.B.:

[MINIMAL\\_user\\_config\\_override.h.zip](#)

[UNIVERSAL\\_user\\_config\\_override.h.zip](#)

oder eine, die man schon **zuvor** bearbeitet hat

(in `\sonoff` befindet sich die Beispiel-Datei: `user_config_override_sample.h`)

2. **Einstellungen** in der [user\\_config\\_override.h](#) bei Bedarf auf **eigene Konfiguration** ändern durch

- **Einfügen / Überschreiben** von Daten z.B. `#define STA_SSID1 ""` zu `#define STA_SSID1 "MeineRouterSSID"`

- **DEAKTIVIEREN** von einzelnen Funktionen und Sensoren durch Löschen von `//` (Kommentar aufheben) vor `#undef`

- **AKTIVIEREN** / Ausklammern durch Setzen von Zeilen als Kommentar durch `//` davor, also `//#undef` oder

durch die Kombination `#undef` und `#define`

Weitere Infos und Erklärungen in den Kommentaren der [user\\_config\\_override.h](#)

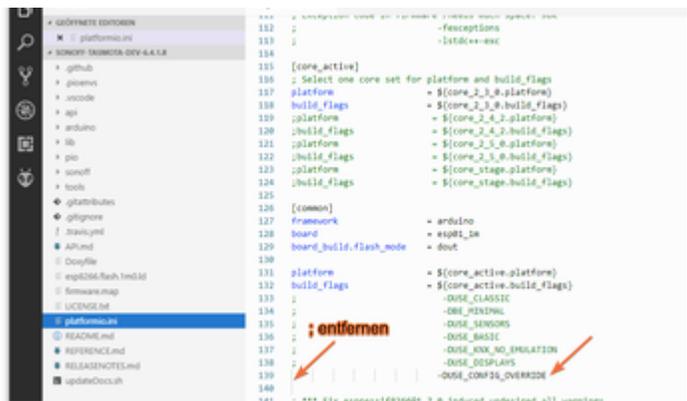
3. [user\\_config\\_override.h](#) aktivieren und [Tasmota](#) kompilieren

Im Normalfall wird die [user\\_config\\_override.h](#) für [Tasmota](#) nicht berücksichtigt,

deshalb müssen vor dem Kompilieren folgende Einträge gemacht werden:

Bei **Atom / VSC** in der `platformio.ini` eine Zeile hinzufügen (bzw. auskommentieren):

`-DUSE_CONFIG_OVERRIDE`



Für **Arduino-IDE** in der [my\\_user\\_config.h](#) folgende Zeile aktivieren (`//` davor löschen):

`#define USE_CONFIG_OVERRIDE`

```
You should have received a copy of the GNU General Public License
along with this program. If not, see <http://www.gnu.org/licenses/>.
*/

#ifndef MY_USER_CONFIG_H
#define MY_USER_CONFIG_H

.....
* This file consists of TWO sections.
*
* SECTION 1:
* The first section contains PARAMETERS overriding flash settings if define CPU_HOLDER is CHANGED.
* All parameters can be persistent changed online using commands via MQTT, WebConsole or Serial.
* Corresponding MQTT/Serial/Console commands are shown in [brackets]
*
* SECTION 2:
* The second section contains Enabled and Disabled FEATURES allowing different program size.
* Changes in this section do NOT need a change of define CPU_HOLDER.
*
* ATTENTION: // davor löschen
* Users are advised to use the user_config_override.h file for most changes.
.....
#define USE_CONFIG_OVERRIDE // Document to use user_config_override.h file. See README.md
.....
* SECTION 1
* - After initial load any change here only take effect if CPU_HOLDER is changed too
.....
// -- Master parameter control! .....
```

### III - FAQ

#### III.I - Welche Einstellungen sind vordefiniert?

[MINIMAL\\_user\\_config\\_override.h.zip](#)

##### Abschnitt 1:

NUR das Wichtigste für WLAN, MQTT, Zeit, Ort, Namen, Sprache...

##### Abschnitt 2:

Default werden die Features so übernommen, wie in der [my\\_user\\_config.h](#) vordefiniert.

Am ENDE kann 'Optionale Firmware Konfiguration' gewählt werden,

dadurch ergibt sich z.B. mit #define USE\_BASIC eine 'kleine' [Tasmota](#) firmware.bin OHNE Sensoren.

Code

```
/*
user_config_override.huserconfigurationoverridesmy_user_config.hforSonoff-Tasmota
```

Copyright (C) 2019

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

```
*/
```

```
#ifndef _USER_CONFIG_OVERRIDE_H_
#define _USER_CONFIG_OVERRIDE_H_
```

```
// force the compiler to show a warning to confirm that this file is included
#warning **** user_config_override.h: Using Settings from this File ****
```

```
/*
```

```
* To modify the stock configuration without changing the my_user_config.h file:
* (1) copy this file to "user_config_override.h" (It will be ignored by Git)
* (2) define your own settings
* (3)
```

```
* define USE_CONFIG_OVERRIDE as
* iel : export PLATFORMIO_BUILD_FLAGS='-DUSE_CONFIG_OVERRIDE'
```

```
* enable define USE_CONFIG_OVERRIDE in
*****
```

```
ChangesSECTIONPARAMETERdefineswilonlyoverrideflashsettingsifouchangeCFG HOLDER.
* - Expect compiler warnings when no ifdef/undef/endif sequence is used.
* - You still need to update my_user_config.h for major define USE_MQTT_TLS.
Allparameterscanbe persistentchangedonlineusing commands viaMQTT,WebConsoleorSerial.
\*****
```

```
//
// +++ Aenderungen erst ab hier vornehmen.
//
// +++ _MINIMAL_user_config_override.h
// +++ NoitaercX +++ 13.01.2019 16:00:00
//
// +++ Kommentare von mir werden durch '+++' gekennzeichnet. +++
// +++ ALLES was hier NICHT geaendert werden soll,
// +++ kann als Kommentar // oder /* */ auskommentiert werden +++
// +++ und bleibt dann so, wie in der my_user_config.h definiert.+++
//
//
// +++ Abschnitt 1 (SECTION 1)
//
// +++ Eigene WLAN-Einstellungen
#undef STA_SSID1
#define STA_SSID1 ""
#undef STA_PASS1
#define STA_PASS1 ""
```

```
// -- Project -----
https://forum.creationx.de/topic/index.php?entry/41-tasmota-anpassung-durch-user-config/ 5
#definePROJECT"meinsonoff"/PROJECTisusedasthedefaulttopicdelimiter
```

```
// +++ Eigene MQTT-Einstellungen
```

Alles anzeigen

[\\_UNIVERSAL\\_user\\_config\\_override.h.zip](#)

#### Abschnitt 1:

ALLES Wichtige für WLAN, MQTT, Zeit, Ort, Namen, Sprache usw. ist aus-/abwählbar.

#### Abschnitt 2:

Eine Grundauswahl mit einigen, der wichtigsten, Sensoren ist voreingestellt.

### III.II - Kann ich meine bearbeitete user\_config\_override.h wiederverwenden?

Ja und uneingeschränkt, ohne weitere Veränderungen, vom einen \sonoff Ordner zum nächsten (Punkt 3. [user\\_config\\_override.h aktivieren](#) muss erneut durchgeführt werden).

### III.III - Warum immer paarweise #undef...#define ?

Wenn ein Wert bereits definiert wurde, muss er vorerst auf 'nicht definiert' gesetzt werden, um ihn neu festzulegen --> bei Mehrfach-Definitionen kommt es sonst zu Fehlern beim Kompilieren.

### III.IV - Komplexere Bedingungen bildet man mit:

```
#ifdef ... // +++ falls definiert ...  
    #undef ... // +++ hebe Definition auf ...  
    #define ... // +++ definiere ...  
#endif  
  
#ifndef ... // +++ falls nicht definiert ...  
    #define ... // +++ definiere ...  
#endif
```

Weitere Beschreibungen und Fragen --> [Auskommentieren der my\\_user\\_config.h](#)

#### P.S.:

Kommentare zu **Korrekturen** und **Verbesserungsvorschlägen** sind SEHR willkommen und werden von mir mit Sicherheit überdacht.

Wenn dir der Beitrag geholfen hat, freut mich ein  natürlich besonders !)