Deutsch • English

API Keys

Key	R/W	Type	Category	Description
rfb	R	int	Status	Relay Feedback
rst	W	any	Other	Ladestation neustarten
alw	R	bool	Status	Darf das Auto derzeit laden?
acu	R	int	Status	Mit wie vielen Ampere darf das Auto derzeit laden?
adi	R	bool	Status	Wird der 16A Adapter benutzt? Limitiert den Ladestrom auf 16A
dwo	R/W	optional<double>	Config	Lade Energy Limit, gemessen in Wh, null bedeutet deaktiviert, nicht mit der Next-Trip Energie zu verwechseln
tpa	R	float	Status	30 Sekunden Gesamtleistungsdurchschnitt (wird für genauere next-trip vorhersagen berechnet)

Key	R/W	Type	Category	Description
sse	R	string	Constant	serial number
eto	R	uint64	Status	energy_total, measured in Wh
wifis	R/W	array	Config	WiFi Konfiguration mit SSID und Passwort; Wenn man nur den zweiten Eintrag ändern möchte, einfach das erste Objekt leer lassen: `[{}, {"ssid":"","key":""}]`
delw	W	uint8	Other	set this to 0-9 to delete sta config (erases ssid, key, ...)
scan	R	array	Status	wifi scan result (encryptionType: OPEN=0, WEP=1, WPA_PSK=2, WPA2_PSK=3, WPA_WPA2_PSK=4, WPA2_ENTERPRISE=5, WPA3_PSK=6, WPA2_WPA3_PSK=7)
scaa	R	milliseconds	Status	wifi scan age
wst	R	uint8	Status	WiFi STA status (IDLE_STATUS=0, NO_SSID_AVAIL=1, SCAN_COMPLETED=2, CONNECTED=3, CONNECT_FAILED=4, CONNECTION_LOST=5, DISCONNECTED=6, CONNECTING=8, DISCONNECTING=9, NO_SHIELD=10 (for compatibility with WiFi Shield library))
wsc	R	uint8	Status	WiFi STA error count
wsm	R	string	Status	WiFi STA error message
wsms	R	uint8	Status	WiFi state machine state (None=0, Scanning=1, Connecting=2, Connected=3)
ccw	R	optional<object>	Status	Currently connected WiFi
wfb	R	array	Status	WiFi failed mac addresses
wcb	R	string	Status	WiFi current mac address
wpb	R	array	Status	WiFi planned mac addresses

Key	R/W	Type	Category	Description
nif	R	string	Status	Default route
dns	R	object	Status	DNS server
host	R	optional<string>	Status	hostname used on STA interface
rssi	R	optional<int8>	Status	RSSI signal strength
tse	R/W	bool	Config	time server enabled (NTP)
tsss	R	uint8	Config	time server sync status (RESET=0, COMPLETED=1, IN_PROGRESS=2)
tof	R/W	minutes	Config	timezone offset in minutes
tds	R/W	uint8	Config	timezone daylight saving mode, None=0, EuropeanSummerTime=1, UsDaylightTime=2
utc	R/W	string	Status	utc time
loc	R	string	Status	local time
led	R	object	Status	internal infos about currently running led animation
lbr	R/W	uint8	Config	led_bright, 0-255
lmo	R/W	uint8	Config	logic mode (Default=3, Awattar=4, AutomaticStop=5)
ama	R/W	uint8	Config	ampere_max limit
clp	R/W	array	Config	current limit presets, max. 5 entries
bac	R/W	uint8_t	Config	Button allow Current change (0=AlwaysLock, 1=LockWhenCarIsConnected, 2=LockWhenCarIsCharging, 3=NeverLock)
sdp	R/W	uint8_t	Config	Button Allow Force change (0=AlwaysLock, 1=LockWhenCarIsConnected, 2=LockWhenCarIsCharging,

Key	R/W	Type	Category	Description
				3=NeverLock)
lbp	R	milliseconds	Status	lastButtonPress in milliseconds
amp	R/W	uint8	Config	requestedCurrent in Ampere, used for display on LED ring and logic calculations
fna	R/W	string	Config	friendlyName
cid	R/W	string	Config	color_idle, format: #RRGGBB
cwc	R/W	string	Config	color_waitcar, format: #RRGGBB
cch	R/W	string	Config	color_charging, format: #RRGGBB
cfi	R/W	string	Config	color_finished, format: #RRGGBB
ust	R/W	uint8	Config	unlock_setting (Normal=0, AutoUnlock=1, AlwaysLock=2)
lck	R	uint8	Status	Effective lock setting, as sent to Charge Ctrl (Normal=0, AutoUnlock=1, AlwaysLock=2, ForceUnlock=3)
sch_week	R/W	object	Config	scheduler_weekday, control enum values: Disabled=0, Inside=1, Outside=2
sch_satur	R/W	object	Config	scheduler_saturday, control enum values: Disabled=0, Inside=1, Outside=2
sch_sund	R/W	object	Config	scheduler_sunday, control enum values: Disabled=0, Inside=1, Outside=2
nmo	R/W	bool	Config	norway_mode / ground check enabled when norway mode is disabled (inverted)
fsp	R	bool	Status	force_single_phase, this is only the result of the charging logic, if it wishes single force or not at the moment
lrn	W	uint8	Other	set this to 0-9 to learn last read card id

Key	R/W	Type	Category	Description
del	W	uint8	Other	set this to 0-9 to clear card (erases card name, energy and rfid id)
acs	R/W	uint8	Config	access_control user setting (Open=0, Wait=1)
frc	R/W	uint8	Config	forceState (Neutral=0, Off=1, On=2)
rbc	R	uint32	Status	reboot_counter
rbt	R	milliseconds	Status	time since boot in milliseconds
car	R	optional<uint8>	Status	carState, null if internal error (Unknown/Error=0, Idle=1, Charging=2, WaitCar=3, Complete=4, Error=5)
err	R	optional<uint8>	Status	error, null if internal error (None = 0, FiAc = 1, FiDc = 2, Phase = 3, Overvolt = 4, Overamp = 5, Diode = 6, PpInvalid = 7, GndInvalid = 8, ContactorStuck = 9, ContactorMiss = 10, FiUnknown = 11, Unknown = 12, Overtemp = 13, NoComm = 14, StatusLockStuckOpen = 15, StatusLockStuckLocked = 16, Reserved20 = 20, Reserved21 = 21, Reserved22 = 22, Reserved23 = 23, Reserved24 = 24)
cbl	R	optional<int>	Status	cable_current_limit in A
pha	R	optional<array>	Status	phases
wh	R	double	Status	energy in Wh since car connected
trx	R/W	optional<uint8>	Status	transaction, null when no transaction, 0 when without card, otherwise cardIndex + 1 (1: 0. card, 2: 1. card, ...)
fwv	R	string	Constant	FW_VERSION
ccu	R	optional<object>	Status	charge controller update progress (null if no update is in progress)
oem	R	string	Constant	OEM manufacturer
typ	R	string	Constant	Devicetype

Key	R/W	Type	Category	Description
fwc	R	string	Constant	firmware from CarControl
ccrv	R	string	Constant	chargectrl recommended version
lse	R/W	bool	Config	led_save_energy
cdi	R	object	Status	charging duration info (null=no charging in progress, type=0 counter going up, type=1 duration in ms)
lccfi	R	optional<milliseconds>	Status	lastCarStateChangedFromIdle (in ms)
lccfc	R	optional<milliseconds>	Status	lastCarStateChangedFromCharging (in ms)
lcctc	R	optional<milliseconds>	Status	lastCarStateChangedToCharging (in ms)
tma	R	array	Status	temperature sensors
amt	R	int	Status	temperatureCurrentLimit
nrg	R	array	Status	energy array, U (L1, L2, L3, N), I (L1, L2, L3), P (L1, L2, L3, N, Total), pf (L1, L2, L3, N)
modelStatus	R	uint8	Status	Reason why we allow charging or not right now (NotChargingBecauseNoChargeCtrlData=0, NotChargingBecauseOvertemperature=1, NotChargingBecauseAccessControlWait=2, ChargingBecauseForceStateOn=3, NotChargingBecauseForceStateOff=4, NotChargingBecauseScheduler=5, NotChargingBecauseEnergyLimit=6, ChargingBecauseAwattarPriceLow=7, ChargingBecauseAutomaticStopTestLadung=8, ChargingBecauseAutomaticStopNotEnoughTime=9, ChargingBecauseAutomaticStop=10, ChargingBecauseAutomaticStopNoClock=11, ChargingBecausePvSurplus=12,

Key	R/W	Type	Category	Description
				ChargingBecauseFallbackGoEDefault=13, ChargingBecauseFallbackGoEScheduler=14, ChargingBecauseFallbackDefault=15, NotChargingBecauseFallbackGoEAwattar=16, NotChargingBecauseFallbackAwattar=17, NotChargingBecauseFallbackAutomaticStop=18, ChargingBecauseCarCompatibilityKeepAlive=19, ChargingBecauseChargePauseNotAllowed=20, NotChargingBecauseSimulateUnplugging=22, NotChargingBecausePhaseSwitch=23, NotChargingBecauseMinPauseDuration=24, NotChargingBecauseError=26, NotChargingBecauseLoadManagementDoesntWant=27, NotChargingBecauseOcppDoesntWant=28, NotChargingBecauseReconnectDelay=29, NotChargingBecauseAdapterBlocking=30, NotChargingBecauseUnderfrequencyControl=31, NotChargingBecauseUnbalancedLoad=32, ChargingBecauseDischargingPvBattery=33, NotChargingBecauseGridMonitoring=34, NotChargingBecauseOcppFallback=35)
lmsc	R	milliseconds	Status	last model status change
mca	R/W	uint8	Config	minChargingCurrent
awc	R/W	uint8	Config	awattar country (Austria=0, Germany=1)
awp	R/W	float	Config	awattarMaxPrice in ct
awcp	R	optional<object>	Status	awattar current price
ido	R	optional<object>	Config	Inverter data override

Key	R/W	Type	Category	Description
ids	R/W	bool	Other	PvSurPlus Information. z.b.: {"pGrid": 1000., "pPv": 1400., "pAkku": 2000.} pGrid < 0 ==> Einspeisen, pAkku < 0 ==> Batterie laden, pPv > 0 ==> PV Produktion, pPv < 0 ==> Standby. Muß alle 5 Sekunden geschrieben werden. Kann bis 10 Sekunden nach Schreiben gelesen werden. pPv und pAkku sind optional
frm	R	uint8	Config	roundingMode PreferPowerFromGrid=0, Default=1, PreferPowerToGrid=2
fup	R/W	bool	Config	usePvSurplus
awe	R/W	bool	Config	useAwattar
fst	R/W	float	Config	startingPower in watts
fmt	R/W	milliseconds	Config	minChargeTime in milliseconds
att	R/W	seconds	Config	automatic stop time in seconds since day begin, calculation: (hours*3600)+(minutes*60)+(seconds)
ate	R/W	double	Config	automatic stop energy in Wh
ara	R/W	bool	Config	automatic stop remain in aWATTar
acp	R/W	bool	Config	allowChargePause (car compatiblity)
cco	R/W	double	Config	car consumption (only stored for app)
esk	R/W	bool	Config	energy set kwh (only stored for app)
fzf	R/W	bool	Config	zeroFeedin
pgt	R/W	float	Config	pGridTarget in W
sh	R/W	float	Config	stopHysteresis in W
psh	R/W	float	Config	phaseSwitchHysteresis in W

Key	R/W	Type	Category	Description
po	R/W	float	Config	prioOffset in W
zfo	R/W	float	Config	zeroFeedinOffset in W
psmd	R/W	milliseconds	Config	forceSinglePhaseDuration (in milliseconds)
sumd	R/W	milliseconds	Config	simulate unpluging duration (in milliseconds)
mpwst	R/W	milliseconds	Config	min phase wish switch time (in milliseconds)
mptwt	R/W	milliseconds	Config	min phase toggle wait time (in milliseconds)
ferm	R	uint8	Status	effectiveRoundingMode
mmp	R	float	Status	maximumMeasuredChargingPower (debug)
tlf	R	bool	Status	testLadungFinished (debug)
tls	R	bool	Status	testLadungStarted (debug)
atp	R	optional<object>	Status	nextTripPlanData (debug)
lpsc	R	milliseconds	Status	last pv surplus calculation
inva	R	milliseconds	Status	age of inverter data
pgrid	R	optional<float>	Status	pGrid in W
ppv	R	optional<float>	Status	pPv in W
pakku	R	optional<float>	Status	pAkku in W
deltap	R	float	Status	deltaP
pnp	R	uint8	Status	numberOfPhases
deltaa	R	float	Other	deltaA
pvopt_averagePGrid	R	float	Status	averagePGrid

Key	R/W	Type	Category	Description
pvopt_averagePPv	R	float	Status	averagePPv
pvopt_averagePAkku	R	float	Status	averagePAkku
mci	R/W	milliseconds	Config	minimumChargingInterval in milliseconds (0 means disabled)
mcpd	R/W	milliseconds	Config	minChargePauseDuration in milliseconds (0 means disabled)
mcpea	R/W	optional<milliseconds>	Status	minChargePauseEndsAt (set to null to abort current minChargePauseDuration)
su	R/W	bool	Config	simulateUnpluggingShort
sua	R/W	bool	Config	simulateUnpluggingAlways
hsa	R/W	bool	Config	httpStaAuthentication
var	R	uint8	Constant	variant: max Ampere value of unit (11: 11kW/16A, 22: 22kW/32A)
loe	R/W	bool	Config	Load balancing enabled
log	R/W	string	Config	load_group_id
lop	R/W	uint16	Config	load_priority
lof	R/W	uint8	Config	load_fallback
map	R/W	array	Config	load_mapping (uint8_t[3])
upo	R/W	bool	Config	unlock_power_outage
pwm	R	uint8	Status	phase wish mode for debugging / only for pv optimizing / used for timers later (Force_3=0, Wish_1=1, Wish_3=2)
lfspt	R	optional<milliseconds>	Status	last force single phase toggle
fsptws	R	optional<milliseconds>	Status	force single phase toggle wished since
spl3	R/W	float	Config	threePhaseSwitchLevel

Key	R/W	Type	Category	Description
psm	R/W	uint8	Config	phaseSwitchMode (Auto=0, Force_1=1, Force_3=2)
oct	W	string	Other	firmware update trigger (must specify a branch from ocu)
ocu	R	array	Status	list of available firmware branches
cwe	R/W	bool	Config	cloud websocket enabled"
cus	R	uint8	Status	Cable unlock status (Unknown=0, Unlocked=1, UnlockFailed=2, Locked=3, LockFailed=4, LockUnlockPowerout=5)
ffb	R	uint8	Status	lock feedback (NoProblem=0, ProblemLock=1, ProblemUnlock=2)
fhz	R	optional<float>	Status	Stromnetz frequency (~50Hz) or 0 if unknown
loa	R	optional<uint8>	Status	load balancing ampere
lot	R/W	uint32	Config	load balancing total amp
loty	R/W	uint8	Config	load balancing type (Static=0, Dynamic=1)
cards	R/W	array	Config	array mit nuter karten daten (name, energiezähler, aktivierungsstatus)
ocppe	R/W	bool	Config	OCPP enabled
ocppu	R/W	string	Config	OCPP server url
ocppg	R/W	bool	Config	OCPP use global CA Store
ocppcn	R/W	bool	Config	OCPP skipCertCommonNameCheck
ocppss	R/W	bool	Config	OCPP skipServerVerification
ocpps	R	bool	Status	OCPP started
ocppc	R	bool	Status	OCPP connected

Key	R/W	Type	Category	Description
ocppca	R	null or milliseconds	Status	OCPP connected (timestamp in milliseconds since reboot) Subtract from reboot time (rbt) to get number of milliseconds since connected
ocppa	R	bool	Status	OCPP connected and accepted
ocppaa	R	null or milliseconds	Status	OCPP connected and accepted (timestamp in milliseconds since reboot) Subtract from reboot time (rbt) to get number of milliseconds since connected
ocpph	R/W	seconds	Config	OCPP heartbeat interval (can also be read/written with `GetConfiguration` and `ChangeConfiguration`)
ocppi	R/W	seconds	Config	OCPP meter values sample interval (can also be read/written with `GetConfiguration` and `ChangeConfiguration`)
ocppai	R/W	seconds	Config	OCPP clock aligned data interval (can also be read/written with `GetConfiguration` and `ChangeConfiguration`)
ocppd	R/W	string	Config	OCPP dummy card id (used when no card has been used and charging is already allowed / starting)
ocppr	R/W	bool	Config	OCPP rotate phases on charger
ocpple	R	string or null	Status	OCPP last error
ocpplea	R	null or milliseconds	Status	OCPP last error (timestamp in milliseconds since reboot) Subtract from reboot time (rbt) to get number of milliseconds since connected
ocpprl	R/W	bool	Config	OCPP remote logging (usually only enabled by go-e support to allow debugging)
ocppck	R/W	string	Config	OCPP client key
ocppcc	R/W	string	Config	OCPP client cert

Key	R/W	Type	Category	Description
ocppsc	R/W	string	Config	OCPP server cert