

Application Programming Interface (API)

Direct loads control

man_load_set[i]	bit	R/W	Managed load state; 0=off, 1=on; i = 10..17
man_analog_out[i]	int	R/W	Analog output value in % i = 10

Timetable loads control

Writes 'Once' and/or 'Recurring' actions to in consumer timetables defining consumer's behavior at specific time intervals.

tt_load_set[i]	bit	R/W	'Once' and 'Recurring' actions; i = 10..17
tt_analog_out[i]	int	R/W	Set analog output value in %; i = 10

Cloud Optimization (Consumers timetable)

Cloud optimization allows writes from Cloud/Edge to consumer timetables, changing consumer's behavior during specific time periods (e.g. Smart Grid functionality).

Following table and use cases below explain the usage of variables in different scenarios.

Variables	Description
dr_load_set[10..17]	Direct load control: 0=off, 1=on.
timetable_cloud_once[0..7]	Enables executing 'Once' actions from Cloud: 1=enabled.
timetable_cloud_recurring[0..7]	Enables executing 'Recurring' actions from Cloud: 1=enabled.
timetable_cloud_analog[0]	Enables writing analog values from Cloud to timetable: 1=enabled.
optimization_command[0..50]	cloud timetable optimization command: 1=set once action, 2=set recurring action, 3=set critical peak tariff, 4=set mask, 5=set analog.
optimization_value[0..50]	timetable optimization value (depends of command): -1=clear/clear critical peak tariff, 0=set off/lo tariff, 1=set on/hi tariff, 2=set critical peak tariff, -100..100 set setpoint, -32768=auto).
optimization_start[0..50]	cloud timetable optimization start time 'hhmmw': hh=hour, mm=minute, w=weekday (1..7, 1=monday).
optimization_index[0..50]	timetable optimization index (0..8): 0=broadcast, 1=first consumer timetable, 8=last consumer timetable.
optimization_time[0..50]	timetable optimization period [min.] (max 1440 minutes).

Variable optimization_command triggers timetable manipulation process so make sure to set all other variables before.

Use cases

SCE01 Set once action

Consumer [3] has to be switched off on Wednesday at 16:45 and be in this state for 1 hour.

```

if ( timetable_cloud_once[3] ) { // 4. consumer (idx=3)
  optimization_index[0] = 4; // 4. consumer (idx=3)
  optimization_start[0] = 16453; // hh=16, mm=45, w=3
  optimization_time[0] = 60; // 1 hour
  optimization_value[0] = 0; // set off
  optimization_command[0] = 1; // set once action
};

// Result:
//          ^xxx
//  |---|---|---|---|---|
// ... 14 15 16 17 18 19 ...
    
```

SCE02 Set recurring action

Consumer [7] has to be switched on every Monday@08:00 and switched off every Monday@21:00.

```

if ( timetable_cloud_once[7] ) { // 8. consumer (idx=7)
  // action on
  optimization_index[0] = 8; // 8. consumer (idx=7)
  optimization_start[0] = 8001; // hh=08, mm=00, w=1
  optimization_time[0] = 15; // only one action
  optimization_value[0] = 1; // set on
  optimization_command[0] = 2; // set recurring action
  // action off
  optimization_index[1] = 8; // 8. consumer (idx=7)
  optimization_start[1] = 21001; // hh=21, mm=00, w=1
  optimization_time[1] = 15; // only one action
  optimization_value[1] = 2; // set off
  optimization_command[1] = 2; // set recurring action
};

// Result:
//          ^
//          v
//  |---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
//  |---|---|---|---|---|---|---|
// ... 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16
// 17 18 19 20 21 22 23 ...
    
```

SCE03 Set mask

Consumer [0] has to be switched on/off following this pattern: Saturday

- @11:15 on
- @11:45 off
- @12:00 on
- @13:30 off
- @13:45 on
- @14:00 off

```

if ( timetable_cloud_once[0] ) { // 1. consumer (idx=0)
    
```

```

optimization_index[0] = 1;      // 1. consumer (idx=0)
optimization_start[0] = 11156;  // hh=11, mm=15, w=6
optimization_time[0] = 180;     // from 11:15 to 14:00 + 15min = 180
minutes
optimization_value[0] = 1531;   // mask: 2#010111111011
optimization_command[0] = 4;    // set mask
};

// Result:
//
//                                     lsb   <--   msb
//                                     110111111010
//
//                                     ^x^x^xxxxx^x^
//
// |---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
// |---|---|---|---|---|---|
// 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17
// 18 19 20 21 22 23

```

SCE04 Set analog

Consumer [0] is the only consumer with analog output. Analog value range is 0..100%.
From Tuesday@14:00 until Tuesday@15:00 set output to 75%.

```

if ( timetable_cloud_analog[0] ) { // 1. consumer (idx=0)
optimization_index[0] = 1;      // 1. consumer (idx=0)
optimization_start[0] = 14002;  // hh=14, mm=00, w=2
optimization_time[0] = 60;     // 60 minutes
optimization_value[0] = 75;    // 75%
optimization_command[0] = 5;   // set analog
};

// Result:
// _____ -----
//
// |---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
// |---|---|---|---|---|---|
// 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17
// 18 19 20 21 22 23

```

SCE05 Set tariff

Set high tariff (HT) from 06:00 until 22:00, for Friday.

```

optimization_start[0] = 6005;   // hh=06, mm=00, w=5
optimization_time[0] = 960;    // only one action
optimization_value[0] = 1;     // high tariff
optimization_command[0] = 3;   // set tariff

// Result:
// _____ -----
// -----
//

```

```
// |---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
-|---|---|---|---|---|
// 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17
18 19 20 21 22 23
```

SCE06 Delete actions from timetable

Clear actions made in SCE01.

```
if ( timetable_cloud_once[3] ) { // 4. consumer (idx=3)
  optimization_index[0] = 4; // 4. consumer (idx=3)
  optimization_start[0] = 16453; // hh=16, mm=45, w=3
  optimization_time[0] = 60; // 1 hour
  optimization_value[0] = -1; // clear
  optimization_command[0] = 1; // set once action
};
```

```
// Result:
//
// |---|---|---|---|---|
// ... 14 15 16 17 18 19 ...
```

From: <http://wiki.hiq-universe.com/> -

Permanent link: <http://wiki.hiq-universe.com/doku.php?id=en:hems:universe:api&rev=1594217970>

Last update: 2020/07/08 14:19

