

# QUICK START STEPS



## Register Device

Login to Sigfox Backend.  
Click New device.  
Create new device using the Device ID, PAC and your selected device type.

**Device - New**

Device information
Identifier (hex) 0000
Name
PAC
End product certificate
Where can I find the end product certificate?
Type datalogger
Lat (-90° to +90°) 0.0
Lng (-180° to +180°) 0.0
Map Locate on map
Subscription automatic renewal <input checked="" type="checkbox"/>
Activable <input checked="" type="checkbox"/>
Ok Cancel

## Set Downlink

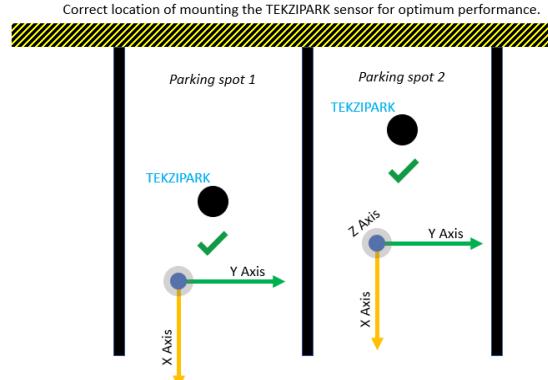
Edit the selected device type.  
Select DIRECT downlink mode and update the downlink data in hexa.  
Select "Custom grammar" for payload parsing and use  
"Occupied::bool:7 Keep-Alive::bool:0  
Reset::bool:1 No\_Beacon::bool:2  
Radar::bool:3 Obstruction::bool:4  
Good\_Battery::bool:5  
Temperature:int:8 Parking\_ID:uint:8  
DEFx:int:8 DEFy:int:8 DEFz:int:8  
Bx:int:8 By:int:8 Bz:uint:8  
Fault\_Code:uint:8 Obst\_Val:uint:8  
D/Reflect:uint:8" for custom configuration

**Device type teczip\_5574\_8f83 - Edition**

Device type information
Name teczip_5574_8f83
Description Testing for TEKZIPARK. This is
Keep-alive (in minutes) 240
Subscription automatic renewal <input checked="" type="checkbox"/>
Contracts teczip_5574_8f83 (no token left - geoloc: no, end date: 2021-03-25)
If we fail to call one of your callbacks, an email will be sent to the address below so that you can take action to fix the problem.
Alert email
Downlink data
Downlink mode DIRECT <input checked="" type="radio"/> For more details on Downlink modes, please refer to documentation.
Expression must either include hexadecimal encoded bytes (e.g. \deadbeef00) or the following variables: {time} 4 bytes - (epoch) 4 bytes - (lat) 4 bytes - (lon) 2 bytes - (ranging) 1 byte
Downlink data in hex 47164061500868AA
Payload display
Payload parsing Custom grammar
Custom configuration Occupied bool 7 Temp 1 int 8 DEFx int 8 RILy int 8
Ok Cancel

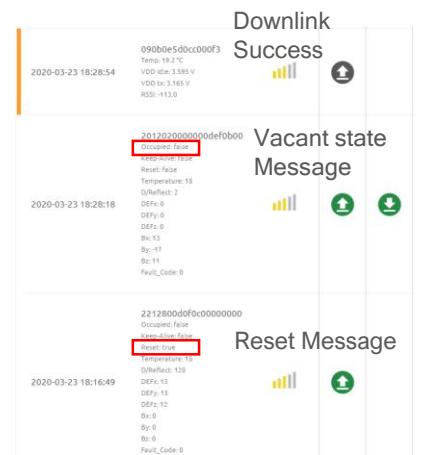
## Install

Select your desired location.  
Bolt down the device with the bolts provided OR drill a hole in the ground for flush version.



## Magnetic Reset

Place a strong magnet over the device.  
Wait for the reset message to appear in the device messages screen.  
Wait for the downlink to finish.  
Device is now operational.



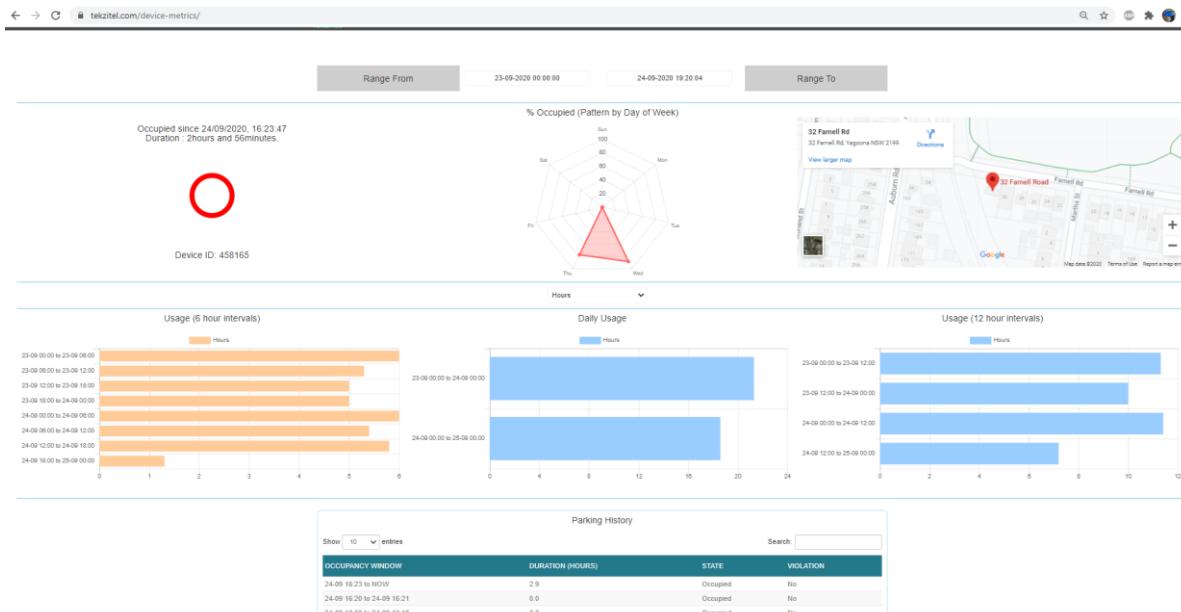
By default the device starts in debug mode, so that it is able to send sensor parametric information to help visualize device behaviour. Debug mode can be disabled via configuration of the device via a downlink message.

# QUICK START STEPS



TEKZITEL provides a miniature visualization utility that can help you to see the parking spot usage where the evaluation devices are installed. This can help as a showcase for customers.

The visualization is at <https://tekzitel.com/dashlanding>.



In order that you can utilise the above feature, a callback needs to be created for the device type under which the evaluation devices are registered as below.

1. Create a custom callback on the Sigfox backend
2. In Custom Payload Config enter -> `Occupied::bool:7`  
`KeepAlive::bool:0 Reset::bool:1 No_Beacon::bool:2`  
`Radar::bool:3 Obstruction::bool:4 Good_Battery::bool:5`  
`Temp:1:int:8 Parking_ID:uint:8`
3. In URL pattern enter -> <https://tekzitel.com/test.php>
4. HTTP Method -> POST
5. Content type -> application/json
6. Body

```
{  
    "data" : "{data}",  
    "time" : "{time}",  
    "sensor_id" : "{device}",  
    "seqNumber" : "{seqNumber}",  
    "occupied" : "{customData#Occupied}",  
    "reset" : "{customData#Reset}",  
    "keepAlive" : "{customData#KeepAlive}",  
    "No_Beacon" : "{customData#No_Beacon}",  
    "Radar" : "{customData#Radar}",  
    "Obstruction" : "{customData#Obstruction}",  
    "Good_Battery" : "{customData#Good_Battery}",  
    "Parking_ID" : "{customData#Parking_ID}"  
}
```

The configuration shows:

- Type: DATA
- Channel: URL
- Custom payload config:  
`Occupied: bool:7`  
`KeepAlive: bool:0`  
`Reset: bool:1`  
`No_Beacon: bool:2`  
`Radar: bool:3`  
`Obstruction: bool:4`  
`Good_Battery: bool:5`  
`Temp: int:8`  
`Parking_ID: uint:8`
- URL pattern: <https://tekzitel.com/test.php>
- Use HTTP Method: POST
- Send SNI: checked
- Headers: header value
- Content type: application/json
- Body:  

```
{  
    "data" : "",  
    "time" : "",  
    "sensor_id" : "",  
    "seqNumber" : "",  
    "occupied" : "true",  
    "reset" : "false",  
    "keepAlive" : "false",  
    "No_Beacon" : "false",  
    "Radar" : "false",  
    "Obstruction" : "false",  
    "Good_Battery" : "false",  
    "Temp" : 0,  
    "Parking_ID" : 0  
}
```