

Car Sharing Solution

[Main Page](#) > [General Information](#) > [Usage scenarios](#) > **Car Sharing Solution**



Contents

- [1 Introduction](#)
- [2 Solution description](#)
- [3 What you need for a solution?](#)
- [4 Installation](#)
- [5 Configuration](#)
 - [5.1 1. Prerequisites:](#)
 - [5.1.1 1.1. Read through First start guide](#)
 - [5.1.2 1.2. Understanding of possible Sleep modes.](#)
 - [5.2 2. Car sharing configuration:](#)
- [6 Parsing information](#)
 - [6.1 1. Prerequisites:](#)
 - [6.1.1 1.1. Open TCP/UDP port](#)
 - [6.1.2 1.2. Read Java parser first start guide](#)
 - [6.2 2. Parsing example:](#)

Introduction

Recently, we have seen the development of car-sharing services used by private individuals, companies, and public institutions alike. These client groups, even though different, all seek comfortable and efficient use of vehicles. Reasons to use such services include fleet optimization, convenience, a sudden need for vehicles, business trips, trips abroad, etc. Also, such matters as car insurance, tire change, maintenance, or car wash are taken care of for the client. We're glad that you have decided to test our "Car Sharing" solution.

Solution description

This solution allows vehicles to be used in car-sharing businesses by monitoring vehicles location and other useful parameters from the vehicle's CAN-BUS such as fuel level, speed, odometer, etc. The most important feature is the ability to lock and unlock the vehicle on demand without a key which is an essential feature for car-sharing businesses. To use this solution FMX1XY device paired with a CAN-CONTROL adapter is needed to allow data to be read from the vehicle and most importantly for door locking and unlocking to be accessible remotely via smartphone or server.

What you need for a solution?

- **FMx1YX** device ([FMB130](#) is taken as example) which is used to connect to a CAN-CONTROL adapter.

Other devices that work with this solution:

[FMB110](#), [FMB120](#), [FMB122](#), [FMB125](#), [FMU125](#), [FMC125](#), [FMM125](#), [FMU130](#), [FMC130](#), [FMM130](#), [FMU126](#)

- The [SIM card](#) in order to get data to your server
- [CAN-CONTROL adapter](#). The adapter allows you to control car doors and it works with many of our products
- [FOTA WEB](#) to remotely send the configuration to the device.

Installation

It's important to hide the tracker, so it would not be a simple task for the thieves to find and unplug it. But also, please do not forget to follow [mounting recommendations](#) as well.

Although devices have high gain antennas it's important to mount devices with stickers on top and in metal-free space. The device should be firmly fixed to the surface or cables. Please make sure, that device is not fixed to heat emitting or moving parts.

The main accessory for the solution is **CAN-CONTROL adapter**, please make sure that the right contacts are connected. Also, check more detailed information [here](#) and [supported vehicle list](#).

To have a working solution it's important to properly wire the devices. In the picture below it's shown how to correctly wire the devices.



During installation please follow recommendations in order to **avoid damaging** device and vehicle:

- Wires should be connected while the module is **not plugged** in.
- Be sure that after the car computer falls asleep, power is still available on the chosen wire. Depending on the car, this may happen in a 5 to 30 minutes period.
- When the module is connected, be sure to measure the voltage again if it did not decrease.
- The ground wire is connected to the vehicle frame or metal parts that are fixed to the frame.

Configuration

1. Prerequisites:

1.1. Read through [First start guide](#)

1.2. Understanding of possible [Sleep modes](#).

2. Car sharing configuration:



Parameter ID - Parameter name GPRS settings:

- [2001](#) - APN
- [2002](#) - APN username (if there are no APN username, empty field should be left)
- [2003](#) - APN password (if there are no APN password, empty field should be left)



Server settings:

- 2004 - Domain
- 2005 - Port
- 2006 - Data sending protocol (0 - TCP, 1 - UDP)

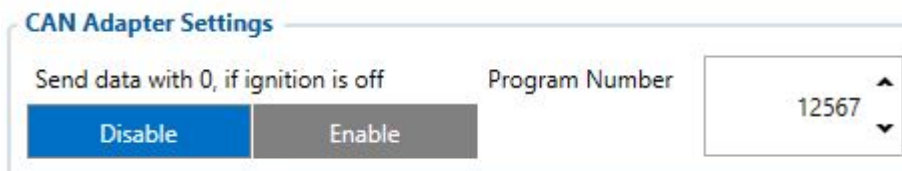
After successful GPRS/SERVER settings configuration, FMB130 device will **synchronize time** and **update records to the configured server**. Time intervals and default I/O elements can be changed by using [Teltonika Configurator](#) or [SMS parameters](#).



Sleep settings:

- 102 - Sleep settings(0 - Disable, 1 - Gps sleep, 2 - Deep sleep, 3 - Online Deep sleep, 4 - Ultra sleep)

Note: This scenario will not work with [Deep Sleep](#) and [Ultra Sleep](#) modes, since they disable the device's GSM module to save power.



The screenshot shows a window titled "CAN Adapter Settings". It contains a toggle switch for "Send data with 0, if ignition is off" with "Disable" selected. To the right is a "Program Number" field with a dropdown menu showing the value "12567".

Can adapter settings:

- 45002 - CAN Adapter program number (This code is written in every connection diagram and it is specific for every vehicle model and make year)



- 45170 - Door status, Priority "Low" (0 - Disabled 1 - Low 2 - High 3 - Panic)



- 45130 - Fuel level, Priority "Low" (0 - Disabled 1 - Low 2 - High 3 - Panic)



- 45220 - Total milage (Counted), Priority "Low" (0 - Disabled 1 - Low 2 - High 3 - Panic)



- 45430 - Security State Flags, Priority "Low" (0 - Disabled 1 - Low 2 - High 3 - Panic)

Quickstart: From default configuration to Car sharing solution in one SMS:

```
" setparam  
2001:APN;2002:APN_user;2003:APN_password;2004:Domain;2005:Port;2006:0;102:3;4  
5002:ProgramNumber;45170:1;45130:1;45220:1;45430:1"
```

This SMS will set up your device to report the Car Sharing scenario to the server and the possibility for remote control of vehicle components.

Note: Before SMS text, two space symbols should be inserted if no SMS username or password was set in SMS \ Call settings.

Parsing information

1. Prerequisites:

1.1. Open [TCP/UDP port](#)

1.2. Read Java parser [first start guide](#)

2. Parsing example:

Unparsed received data in hexadecimal stream

```
0000000000000008F080100000177B0A5F83800000000000000000000004400280D0000001F09EF00F0001504B301B40051005200590098000B42307443
0F8244000011001812002D13FFE85400BE5500005A0000970000A800000AF10000601A53000000005703CB6B986400002E
97660000000069023B33856B000065B27B000000007D00000000BD00000000018400800000300000050100005C92
```

AVL Data Packet Part

HEX Code Part

Zero Bytes	00 00 00 00
Data Field Length	00 00 00 8F
Codec ID	08 (Codec 8)
Number of Data 1 (Number of Total Records)	01
Timestamp	00 00 01 77 B0 A5 F8 38 (Thursday, February 18, 2021 3:43:00 PM GMT+02:00)
Priority	00
Longitude	00 00 00 00
Latitude	00 00 00 00
Altitude	00 44
Angle	00 28
Satellites	0D
Speed	00 00
Event IO ID	00
N of Total ID	1F
N1 of One Byte IO	09
1'st IO ID	EF (AVL ID: 239, Name: Ignition)
1'st IO Value	00
2'nd IO ID	F0 (AVL ID: 240, Name: Movement)
2'nd IO Value	00
3'rd IO ID	15 (AVL ID: 21, Name: GSM Signal)
3'rd IO Value	04
4'th IO ID	B3 (AVL ID: 179, Name: Digital Output 1)
4'th IO Value	01
5'th IO ID	B4 (AVL ID: 180, Name: Digital Output 2)
5'th IO Value	00
6'th IO ID	51 (AVL ID: 81, Name: Vehicle Speed)
6'th IO Value	00
7'th IO ID	52 (AVL ID: 82, Name: Accelerator Pedal Position)

7'th IO Value	00
8'th IO ID	59 (AVL ID: 89, Name: Fuel level)
8'th IO Value	00
9'th IO ID	98 (AVL ID: 152, Name: Geofence zone 05)
9'th IO Value	00
N2 of Two Byte IO	0B
1'st IO ID	42 (AVL ID: 66, Name: External Voltage)
1'st IO Value	30 74
2'nd IO ID	43 (AVL ID: 67, Name: Battery Voltage)
2'nd IO Value	0F 82
3'rd IO ID	44 (AVL ID: 68, Name: Battery Current)
3'rd IO Value	00 00
4'th IO ID	11 (AVL ID: 17, Name: Axis X)
4'th IO Value	00 18
5'th IO ID	12 (AVL ID: 18, Name: Axis Y)
5'th IO Value	0F 8A
6'th IO ID	13 (AVL ID: 19, Name: Axis Z)
6'th IO Value	FF E8
7'th IO ID	54 (AVL ID: 84, Name: Fuel level)
7'th IO Value	00 BE
8'th IO ID	55 (AVL ID:85, Name: Engine RPM)
8'th IO Value	00 00
9'th IO ID	5A (AVL ID:90, Name: Door Status)
9'th IO Value	00 00
10'th IO ID	97 (AVL ID:151, Name: Battery Temperature)
10'th IO Value	00 00
11'th IO ID	A8 (AVL ID: 168, Battery Voltage)
11'th IO Value	00 00
N4 of Four Byte IO	0A
1'st IO ID	F1 (AVL ID: 241, Name: Active GSM Operator)
1'st IO Value	00 00 60 1A
2'nd IO ID	53 (AVL ID: 83, Name: Fuel Consumed)
2'nd IO Value	00 00 00 00
3'rd IO ID	57 (AVL ID: 87, Name: Total Mileage)
3'rd IO Value	03 CB 6B 98
4'th IO ID	64 (AVL ID: 100, Name: Program Number)
4'th IO Value	00 00 2E 97
5'th IO ID	66 (AVL ID: 102, Name: Engine Worktime)
5'th IO Value	00 00 00 00
6'th IO ID	69 (AVL ID: 105, Name: Total Mileage (counted))
6'th IO Value	02 3B 33 85
7'th IO ID	6B (AVL ID: 107, Name: Fuel Consumed(counted))
7'th IO Value	00 00 65 B2

